



Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

2002-09

Integrated theater assessment profiling system

Wiest, James A.; Hadley, Michael P.

Monterey, California. Naval Postgraduate School

<http://hdl.handle.net/10945/5154>

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun



Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>

NAVAL POSTGRADUATE SCHOOL Monterey, California



THESIS

INTEGRATED THEATER ASSESSMENT PROFILING SYSTEM

by

Michael P. Hadley
and
James A. Wiest

September 2002

Thesis Co-Advisors:

Alexander Bordetsky
Richard S. Cote'

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2002	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: Integrated Theater Assessment Profiling System			5. FUNDING NUMBERS	
6. AUTHOR(S) Michael Hadley and James A. Wiest				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING/MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) The Integrated Theater Assessment Profiling System (iTAPS) takes the original stove-piped Theater Assessment Profiling System (TAPS) software solution and turns it into a robust, data-centric, web-based decision support system for Commander, Second Fleet. ITAPS uses the .Net Framework and ASP.NET/ADO.NET, along with SQL Server to provide a web-enabled application that gives an overarching, abstracted view of the battle space for the Operational Commander while still providing drill-down capability and trend analysis tools if more detail is desired. The software was developed using the extreme programming technique and black box testing methods. A demonstration was performed at Second Fleet to test its acceptability and usability.				
14. SUBJECT TERMS Decision Support System, .NET Framework, Radar Diagrams, Theater Assessment			15. NUMBER OF PAGES 224	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

INTEGRATED THEATER ASSESSMENT PROFILING SYSTEM

Michael P. Hadley
Captain, United States Marine Corps
B.S., Commerce, University of Virginia, 1991

Submitted in partial fulfillment of the
requirements for the degrees of

**MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT
and
MASTER OF SCIENCE IN COMPUTER SCIENCE**

James A. Wiest
Lieutenant, United States Navy
B.S., Applied Math, University of Idaho, 1995

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

**NAVAL POSTGRADUATE SCHOOL
September 2002**

Authors: Michael P. Hadley

James A. Wiest

Approved by: Alexander Bordetsky
Thesis Co-Advisor

Richard S. Coté
Thesis Co-Advisor

Dan Boger
Chairman, Department of Information Sciences

Chris Eagle
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Integrated Theater Assessment Profiling System (iTAPS) takes the original stove-piped Theater Assessment Profiling System (TAPS) software solution and turns it into a robust, data-centric, web-based decision support system for Commander, Second Fleet. iTAPS uses the .Net Framework and ASP.NET/ADO.NET, along with SQL Server to provide a web-enabled application that gives an overarching, abstracted view of the battle space for the Operational Commander while still providing drill-down capability and trend analysis tools if more detail is desired. The software was developed using the extreme programming technique and black box testing methods. A demonstration was performed at Second Fleet to test its acceptability and usability.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION	1
A.	BACKGROUND	1
B.	PROBLEM STATEMENT	3
1.	Current System Limitations	3
2.	Thesis Scope.....	4
3.	Assumptions	4
4.	Research Questions.....	4
C.	RESEARCH OBJECTIVES.....	5
II.	TECHNOLOGY REVIEW	7
A.	DECISION SUPPORT STRUCTURES	7
B.	DATABASE TOOLS	9
1.	Microsoft Access	9
2.	SQL Server.....	11
C.	PROGRAMMING TOOL COMPARATIVE ANALYSIS	13
4.	Tools Examined.....	15
a.	Active Server Pages (ASP).....	16
b.	Extensible Markup Language (XML).....	16
c.	Flash.....	17
d.	Java.....	18
e.	DAML (DARPA Agent Markup Language)	19
f.	ASP.NET	20
D.	TECHNOLOGY CONCLUSIONS.....	23
III.	THE DECISION SUPPORT SYSTEM.....	25
A.	DSS DESIGN SYNTHESIS	25
1.	Level 3 MDM	25
2.	ODSS	26
B.	INPUTS.....	27
1.	Administrators	27
C.	OUTPUTS.....	28
1.	Output Users	28
2.	Desired Information Displays	28
3.	Archiving and Data Mining	28
D.	USER INTERFACE	28
1.	Current Structures	28
2.	Improvements	29
3.	Alternatives.....	29
D.	USER INTERFACE DESIGN	31
2.	Administrative Pages.....	32
3.	View Pages.....	33
4.	User Feedback Mechanisms.....	33
IV.	METHODOLOGY	35
A.	EVALUATE CURRENT SOFTWARE CAPABILITIES	35

1.	Findings from Research.....	35
2.	What Was Chosen and Why.....	36
B.	DEVELOP WEB-BASED SOFTWARE SOLUTION	38
1.	Architecture for Solution Chosen.....	38
2.	System Design.....	39
a.	Database.....	39
b.	Web Pages	40
c.	Integration with the Application Server	42
4.	System Development.....	43
a.	Development Methodology	43
b.	Test Lab.....	44
c.	Testing Methodology	44
d.	Documentation	45
5.	System Delivery.....	45
V.	FINAL RESULTS	47
A.	SYSTEM CAPABILITIES	47
B.	SYSTEM LIMITATIONS	51
VI.	CONCLUSION.....	55
VII.	RECOMMENDATIONS FOR FURTHER RESEARCH	57
A.	PRESENTATION OF DATA WITH RADAR CHARTS.....	57
B.	ADD TRIP WIRE CAPABILITY	57
C.	ADD DATA MINING FEATURES	57
D.	IMPROVE DRILLDOWN CODING	58
E.	CREATE SCALABLE PAGES	58
F.	CUSTOMIZE FOR DIFFERENT BROWSERS.....	58
G.	AUTOMATE ENTIRE INSTALL	58
H.	IMPORT DOMAIN USERNAMES FOR GRADERS	59
I.	ADAPT TO WORK WITH CODS	59
J.	ADD PICTURE AND VIDEO LINK CAPABILITY FOR GRADE INPUT.....	59
	APPENDIX A ITAPS USERS MANUAL	61
	APPENDIX B DATABASE SCHEMA	207
	INITIAL DISTRIBUTION LIST	209

LIST OF FIGURES

Figure 1.	Current TAPS Architecture	2
Figure 2.	Current TAPS Screen Shot.....	3
Figure 3.	ASP.NET within the .NET Framework	23
Figure 4.	StopLight Chart Example	30
Figure 5.	Color Line Chart Example	30
Figure 6.	iTAPS 3-tiered Architecture	39
Figure 7.	iTAPS Screen Shot	48
Figure 8.	TAPS Grade Detail.....	50
Figure 9.	PS Grade Detail.....	51

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Benefits and Limitations of DSS Use (iTAPS benefits in bold)	8
Table 2.	Level 3 MDM Characteristics (iTAPS in bold)	26
Table 3.	Development Product Comparison for iTAPS Requirements	35

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

This thesis is the direct result of a critical need expressed by the Second Fleet J3 to modernize their Theater Assessment Profiling System (TAPS) software. Second Fleet has developed the current solution in-house to mimic some of the behavior of an expensive commercial product that had been presented to them in the past. Second Fleet like many things about the commercial product, but it was expensive and required a complex installation with special servers and client side code.

The current TAPS uses a stovepipe arrangement of a Microsoft Access Database, Hypertext Markup Language (HTML) web forms and multiple Microsoft Excel spreadsheets for the mathematical functions and display of spider graphs (see Figure 1). These Excel documents are then merged and imported into a Microsoft PowerPoint presentation. Finally, this presentation is then converted to a web document using the “save as web page” functionality of PowerPoint.

This system serves as a Decision Support System (DSS) for the Commander, Second Fleet, giving the Commander and his staff an overall view of functional areas and provides drill-down capability within the presentation. This system provides basic functionality but is not real-time, is manpower intensive and requires a specific set of software tools to be installed on the client machines to facilitate its use.

There are two main “modes” of how TAPS functions. In the first mode, users input data to a central database via the local webspace utilizing the tools in Access for web form creation. At predetermined times the Second Fleet J3 and J6 staff enter the second mode and pull the data from Access to a series of Excel spreadsheets and then update the PowerPoint presentation. Finally, this PowerPoint Presentation is saved as a web page for the various briefings

throughout the day (see Figure 2). The Admiral and his staff then access and analyze the information during these briefings.

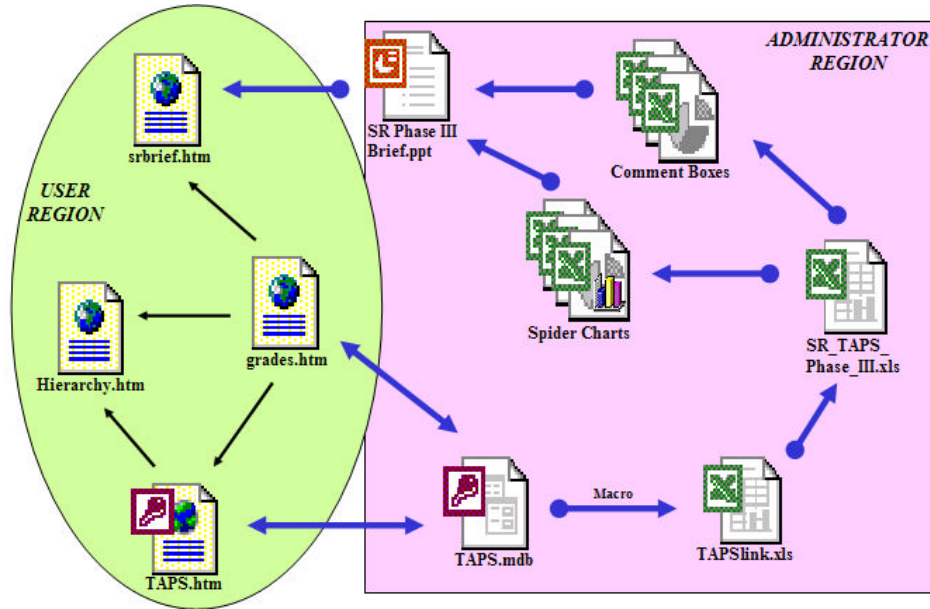


Figure 1. Current TAPS Architecture

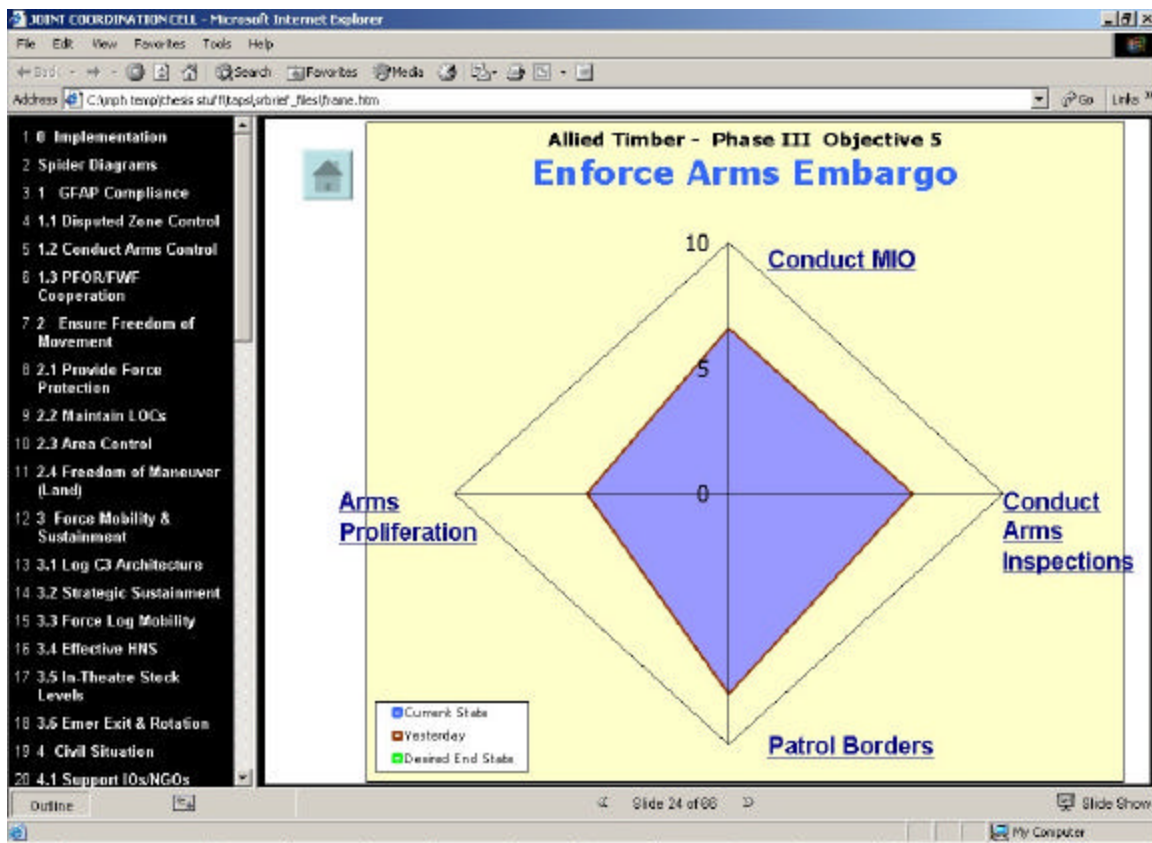


Figure 2. Current TAPS Screen Shot

B. PROBLEM STATEMENT

The current system is limited in its functionality, the age of its data, its performance, and it is of limited use to the Commander, Second Fleet to make timely and informed decisions. It also prevents the Second Fleet staff from maintaining and updating TAPS quickly and effectively.

1. Current System Limitations

- Stove-piped software system architecture. TAPS can not integrate with any other systems.
- Non-real-time update and access.
- Requires specific client-side software architecture. In order to view the web pages created by TAPS, elements of Microsoft Office 2000 must be loaded onto the client machine.
- Limited data security and data compartmentalization.
- Manpower intensive system operation.

2. Thesis Scope

The scope of this thesis is to research and develop an improved, integrated, and web-based theater assessment profiling system for Second Fleet. It answers one question: What is currently the best way to port the current functionality of TAPS to a near-real-time, fully web-integrated application? This will be accomplished by evaluating the current architecture and examining the latest alternatives in Commercial-Off-The-Shelf (COTS) product solutions and constructing the best solution for the Integrated Theater Assessment Profiling System (iTAPS).

3. Assumptions

The following assumptions are stated for clarity:

- What are the needs and requirements of Second Fleet?
- The iTAPS will be used for both training and actual mission environments
- iTAPS will need to be portable to other similar environments (such as the other numbered fleets)
- iTAPS will need to be designed for upgradability
- iTAPS will need full documentation and a single software install solution for the server side
- iTAPS will be designed to require a minimal client-side software environment
- iTAPS will be designed to be scalable
- iTAPS need not include any functionality not found in the current version of TAPS.

4. Research Questions

- What are the options for integrating the current legacy software solution into a single web-based application?
- Do any preexisting software designs support this type of task?
- What is the optimal software architecture to use in developing this application?
- What are the software choice limitations imposed by the Navy and Second Fleet information system policies?
- What are the networking capabilities and limitations provided by the current architecture and communications hardware?

- What are the security requirements and limitations of this system?

C. RESEARCH OBJECTIVES

The Research objectives are as follows:

- Research and develop an optimal web-based solution for porting the TAPS system to a fully web-based, near-real-time DSS
- Examine how this technology could benefit other similar DSS's in the Department of Defense (DOD)
- Make recommendations for future research in this area

THIS PAGE INTENTIONALLY LEFT BLANK

II. TECHNOLOGY REVIEW

There are several technologies that are critical to making the TAPS system web-based. First, the fundamental IT structure of the TAPS system must be defined. Second, a database schema and format must be developed. Finally, an API for developing the web forms and pages needed must be chosen. Of course, these choices must be made reflecting the desires of, while also understanding Second Fleet's expressed needs. There are two primary constraints placed upon iTAPS. First, iTAPS should be completely server-side if at all possible. Second, standardization on Microsoft products is strongly desired. Within these constraints the following Technology areas were researched: Decision Support Structures, Database Tools and Internet Web Tools.

A. DECISION SUPPORT STRUCTURES

Fundamentally, iTAPS is a decision support system (DSS) for the Commander of a Joint Task Force to quickly analyze the current operation and adjust priorities and orders accordingly. As such, a brief description of decision support systems and their theory and use is in order. Gorry and Scott Morton¹ define traditional decision support systems as:

Decision support systems couple the intellectual resources of individuals with the capabilities of the computer to improve the quality of decisions. [They comprise] a computer-based support system for management decision makers who deal with semi-structured problems.

Another descriptive definition that is helpful for iTAPS is made by Marakas²:

A decision support system is a system under the control of one or more decision makers that assist in the activity of decision making by providing an organized set of tools intended to impart structure to portions of the decision making situation and to improve the ultimate effectiveness of the decision outcome.

¹ Gorry, G. A. and M. S. Scott Morton, 1989, "A Framework for Management Information Systems." *Sloan Management Review* 13(1), p. 60

² George M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 4

Both definitions give us insight into the purpose of a DSS and make it clearly evident that iTAPS is a DSS. Specifically iTAPS is a DSS, under the control of COMSECONDFLT, which provides tools to impart structure to the decision-making situation, enabling more effective decisions and ultimately improves the outcome in our favor. It is important to realize that a DSS has certain benefits as well as limitations (See Table 1). The important iTAPS benefits are bolded in Table 1.

<u>Benefits</u>	<u>Limitations</u>
Extend the decision makers ability to process information and knowledge	DSSs cannot yet be defined to contain distinctively human decision-making talents such as creativity, imaginativeness, or intuition
Extend the decision makers ability to tackle large-scale, time-consuming, complex problems	The power of a DSS is limited by the computer system upon which it is running, its design, and the knowledge it possesses at the time of it's use
Shorten the time associated with making a decision	Language and command interfaces are not yet sophisticated enough to allow for natural language processing of user directives and inquiries
Improve the reliability of a decision process or outcome	DSSs are normally designed to be narrow in scope of application, thus inhibiting their generalizability to multiple decision-making contexts
Encourage exploration and discovery on the part of the decision maker	
Reveal new approaches in thinking about a problem space or decision context	
Generate new evidence in support of a decision or confirmation of existing assumptions	
Create a strategic or competitive advantage over competing organizations	

Table 1. Benefits and Limitations of DSS Use³ (iTAPS benefits in bold)

³ George M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 5

Now that it has been established that iTAPS is a DSS, it is important to illustrate that iTAPS is a special type of DSS called an Executive Information System (EIS). An EIS is a type of DSS that is uniquely designed to facilitate the analysis of information critical to the overall operation of an organization and to provide an arsenal of tools that can support the strategic decision-making processes conducted by top executives⁴. Within this paradigm we intend to follow the conventions and characteristics of the EIS DSS to develop iTAPS.

From Marakas'⁵ listing of common features and functions of a modern EIS, it is clear that iTAPS must support the following six features:

- Status access, drill down, exception reporting, trend analysis, and ad hoc queries/reports
- Multiple user interfaces
- Graphical user interface (GUI) navigation
- Data management capabilities
- Multidimensional data mining and visualization
- Multilevel access control security

It is from this baseline we intend to develop the iTAPS DSS framework. In addition, by leaving the system open and flexible within these functions, it should be scalable for any size command or operation, thus expanding its use well beyond Second Fleet.

B. DATABASE TOOLS

There are many excellent database products available. While others, such as Oracle make excellent products, one of Second Fleet's stated desires is to have a Microsoft solution if possible. This seems reasonable, as the Navy has doctrinally standardized on many Microsoft products. Given Second Fleet's desire is to have standardization on Microsoft products, within that constraint there are two database products that are possibly suitable for iTAPS.

1. Microsoft Access

⁴ George M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 185

⁵ George M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 197

The Access database system is Microsoft's low-end database product. It is fairly flexible and robust, and it supports Structured Query Language (SQL) commands. Access uses the Jet database engine⁶, which is a proprietary engine developed by Microsoft. While Access has many usability enhancements such as integrated web form development and a more "plain-language" front end, it suffers from several limitations that make it questionable for use with iTAPS. Microsoft acknowledges that Access (and the Jet database engine) was designed for small, local databases, not large web-based data servers. In fact, they state that⁷:

The Access ODBC Driver and Microsoft OLE DB Provider for Microsoft Jet are not intended to be used with high-stress, high-concurrency, 24x7 server applications, such as web, commerce, transactional, messaging servers, and so on.

Given that iTAPS is intended to be a high-stress, high concurrency database with 24x7 access, the Jet database engine does not appear particularly robust for the task. There are also other significant limitations of the Jet database engine. These limitations are as follows⁸:

- **Supported number of concurrent users.** Access was not designed to support a large number of concurrent users. Microsoft states that:

Jet can support up to 255 concurrent users, but performance of the file-based architecture can prevent its use for many concurrent users. In general, it is best to use Jet for 10 or fewer concurrent users.

- **Scalability.** Because Access can realistically support few users, its scalability is limited.
- **Backup.** There is no integrated backup scheme in Access. You must manually copy the database files to backup your data. In addition, to accomplish the backup, all users must be off of the database, something that is almost impossible to do in a dynamic 24x7 environment.

⁶ <http://support.microsoft.com/default.aspx?scid=kb;EN-US;q222135>

⁷ <http://support.microsoft.com/default.aspx?scid=kb;EN-US;q222135>

⁸ <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnmsde/html/msdeforvs.asp>

- **Replication.** This is related to doing a backup. There is no method for replication in Access except for manually copying the database file. This limits availability if you intend to mirror the database for any reason.
- **Security.** Access only supports protecting the database itself, it provides no tools for everyone to have multiple levels of configurable security. Being able to restrict access by table or object is essential to preserving data integrity within iTAPS.
- **Transactions.** Access is not a true transaction-based database. The only way to roll back an operation is to revert to a previous copy of the database.
- **Triggers.** Access has minimal support for triggering events based on database activity and is inefficient at doing so once you code for it.
- **SQL Mail.** Access provides no way to inform administrators by email if the database has a problem.
- **Jobs.** Access has no support for scheduling and automating tasks. Everything must be invoked by the user. The user must invoke everything.

The bottom line is that Access was designed for a particular type of environment, the small business that wants to maintain its records locally, and not for the enterprise level environment found at Second Fleet. Given these limitations of Access and the Jet database engine, we believe a more robust enterprise level solution is in order.

2. SQL Server

SQL Server is Microsoft's enterprise database product. The structure and Integrated Development Environment (IDE) for SQL is also very similar to Access, thus making a "port" of the current structure of TAPS to SQL a somewhat manageable task.

The most compelling reason for "upgrading" TAPS to SQL Server with iTAPS is that, without re-listing them here, all of the deficiencies listed for Access discussed above are overcome easily with SQL Server. Specifically, it is a transaction-based relational database with robust scalability, accessibility and security. It was designed from the ground up to be a web-enterprise tool, and as

such, fits well into the iTAPS structure. The primary advantages for SQL Server are⁹:

- **Internet Integration.** The SQL Server 2000 database engine includes integrated Extensible Markup Language (XML) support. It also has the scalability, availability, and security features required to operate as the data storage component of the largest Web sites. The SQL Server 2000 programming model is integrated with the Windows Distributed interNet Applications Architecture (DNA) architecture for developing Web applications, and SQL Server 2000 supports features such as English Query and the Microsoft Search Service to incorporate user-friendly queries and powerful search capabilities in Web applications.
- **Scalability and Availability.** The same database engine can be used across platforms ranging from laptop computers running Microsoft Windows® 98 through large, multiprocessor servers running Microsoft Windows 2000 Data Center Edition. SQL Server 2000 Enterprise Edition supports features such as federated servers, indexed views, and large memory support that allow it to scale to the performance levels required by the largest Web sites.
- **Enterprise-Level Database Features.** The SQL Server 2000 relational database engine supports the features required to support demanding data processing environments. The database engine protects data integrity while minimizing the overhead of managing thousands of users concurrently modifying the database. SQL Server 2000 distributed queries allow you to reference data from multiple sources as if it were a part of a SQL Server 2000 database, while at the same time, the distributed transaction support protects the integrity of any updates of the distributed data. Replication allows you to also maintain multiple copies of data, while ensuring that the separate copies remain synchronized. You can replicate a set of data to multiple, mobile, disconnected users, have them work autonomously, and then merge their modifications back to the publisher.
- **Ease of installation, deployment, and use.** SQL Server 2000 includes a set of administrative and development tools that improve upon the process of installing, deploying, managing, and using SQL Server across several sites. SQL Server 2000 also supports a standards-based programming model integrated with the Windows DNA, making the use of SQL Server databases and data warehouses a seamless part of building powerful and scalable systems. These features allow you to rapidly deliver SQL Server

⁹ http://msdn.microsoft.com/library/default.asp?url=/library/en-us/architec/8_ar_00_1lbn.asp

applications that customers can implement with a minimum of installation and administrative overhead.

- **Data warehousing.** SQL Server 2000 includes tools for extracting and analyzing summary data for online analytical processing. SQL Server also includes tools for visually designing databases and analyzing data using English-based questions.

The primary advantage of SQL Server for the iTAPS system is the scalability function. While the current implementation of iTAPS is limited, it is being developed with eye towards the future. It is envisioned as a tool that could have thousands of potential inputs and users. Given the significant advantages of SQL Server and its integration into the Visual Studio development tools, it is a superior database engine choice for the iTAPS environment.

C. PROGRAMMING TOOL COMPARATIVE ANALYSIS

There are literally hundreds of web technologies that have feature-sets that could be used for iTAPS. The issues that are important for iTAPS are software integration, complete system specification support in a single product and prevalence of use.

1. Software System Integration (SSI)

While there are many great stove-piped web applications that could be pulled together to create the iTAPS environment, moving away from integration is a step backwards in web technology development. The entire technological field is moving toward, not away from integration. The Center for Advanced Software Systems Integration at Ohio University states that¹⁰:

First, SSI allows more rapid transfer of data among vendors, manufacturers, and customers thereby increasing communication and shortening lead times. The main advantage of shortened lead times is greater flexibility to changes in the demand profile that need to be propagated through all aspects of the complicated customer-vendor network (rapid response).

Second, SSI allows system design errors to be discovered and eliminated or provided for prior to production, because the sub-system models that have

¹⁰ <http://webise.ent.ohiou.edu/cassi/advantages.html>

been developed by separate vendors with separate software tools can be simulated and evaluated as a cohesive system.

With disparate applications, compatibility and versioning issues are constant issues that must be dealt with. By taking advantage of integration we are reducing software development time and reducing potential compatibility issues. Integrated information technology solutions with flexible and open architecture are a force-multiplier for the Navy, just as they are a business-multiplier in the commercial sector.

2. Complete System Specification Support in a Single Software Package

This is a corollary of integration. There are numerous examples in military procurement where different systems, all ostensibly supporting the entire system specification in aggregate, could not work together when integrated. The Air Force F-111 aircraft electronics suite is one such example. In truth, the interdependent interfaces are the most important structures in the IT development world, which is why products that minimize the number of disparate programs involved provide several advantages:

- Reduced interface requirements
- Minimized compatibility issues
- Standardized debugging
- Standardized Support
- Reduced software development life-cycles

While it is true that sometimes a single product might not provide the cleanest or “favorite” method of accomplishing the task, the advantages normally outweigh the special, proprietary ways you would have to integrate separate pieces of software. A good example is in traditional office software. The primary vendors ship software suites because most users recognize the power of having an integrated tool that covers the feature sets they need. For example: few organizations would purchase WordPerfect for word processing, Lotus 1*2*3 for a spreadsheet and Microsoft Access as their database. They would most likely

choose one company's complete office software package for integration and interoperability issues vice specific favorite features they might prefer in each individual product. The same is true in software development, although this is to a lesser extent, as web applications normally have well-defined interfaces. For example: it is relatively trivial to incorporate a Flash animation, or Java applet into an HTML document for display on a web browser. The problem of interoperability and complete system specification coverage remains, however, because, for this example, Java doesn't talk to Flash natively, and vice versa. As we will see, the Extensible Markup Language (XML) specification goes a long way toward interfacing web applications. Tools like ASP.NET provide robust interfaces for taking advantage of XML's power and flexibility. Software development tools that provide maximum coverage of the iTAPS system specification are highly desired for these reasons.

3. Prevalence of Use

While the military has not considered this aspect in traditional procurement and system development, it has become a very important aspect for software development. Software that is widely implemented multiplies in value to its users, much like the oft-cited example of the fax machine. One fax machine in the world isn't very useful. Two fax machines double the original fax machines value. And so, as the number of fax machines grows, so does the value of each fax machine. The identical process occurs on the Internet. The more widely adopted an application or web technology, the more valuable it becomes. When such technologies become de facto standards, it is important for the military to take advantage of these tools because it almost guarantee's their continued support, and provides some measure of obsolescence protection. Active Server Pages (ASP) is one such example for the web development environment.

4. Tools Examined

The bottom line is that web solutions should be chosen that are standardized, cover the desired system specification, and are as highly integrated as possible. Some of the "Top-Tier" solutions we researched are ASP, XML, Flash, Java, DAML and ASP.NET.

a. Active Server Pages (ASP)

Active Server Pages were an outgrowth of Common Gateway Interface (CGI) that was developed as a standard for interfacing external applications with information servers, such as HTTP or Web servers. CGI was scripting support for running server side code for things that standard HTML couldn't handle. ASP took the functional concept of CGI and improved it. It ran as a service and was multi-threaded. It also defined a common interface for programming ASP pages. Microsoft refers to ASP as¹¹:

Active Server Pages is an open, compile-free application environment in which you can combine HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. Active Server Pages enables server-side scripting for IIS with native support for both VBScript and JScript.

So ASP is still basically a scripting tool within HTML. ASP is also widely used on the Internet.¹² It enables a powerful set of features including database access and web form tools in addition to access to COM (Component Object Model) and DCOM (Distributed Component Object Model) objects, both very powerful object-oriented programming tools. ASP is ideally suited to the iTAPS task as it meets our stated criteria quite well. The main disadvantage of ASP is that it requires knowledge of several scripting languages to use and maintain, and its feature set has been significantly upgraded by ASP.NET, Microsoft's newest web development environment.

b. Extensible Markup Language (XML)

XML is a software technology that has taken the Internet by storm since its introduction in 1998. The power of XML is in its flexibility and dynamic configurability. Unlike HTML (Hypertext Markup Language) which has global, specific syntax, XML has a user definable syntax that is infinitely variable. The Extensible Markup Language (XML) 1.0 (Second Edition) definition¹³ states:

¹¹ www.microsoft.com

¹² www.netcraft.com/survey

¹³ Extensible Markup Language (XML) 1.0 (Second Edition) document, W3C Recommendation 6 October 2000

Extensible Markup Language, abbreviated XML, describes a class of data objects called XML documents and partially describes the behavior of computer programs which process them. XML is an application profile or restricted form of SGML, the Standard Generalized Markup Language [ISO 8879]. By construction, XML documents are conforming SGML documents.

XML documents are made up of storage units called entities, which contain either parsed or unparsed data. Parsed data is made up of characters, some of which form character data, and some of which form markup. Markup encodes a description of the document's storage layout and logical structure. XML provides a mechanism to impose constraints on the storage layout and logical structure.

So the power of XML is in its self-definability. In fact, several of the other protocols and programming tools we discuss below take advantage of and make use of XML for their underlying structures. Flash and ASP.NET in particular, build on the XML base to make powerful, content rich web applications. As such, those technologies have several advantages over “straight” XML, which makes them more attractive for the iTAPS DSS implementation. Specifically, Flash and ASP.NET provide automated coding techniques and Graphical User Interface (GUI) Integrated Development Environments (IDEs) that make the XML learning curve less steep, as well as providing more rapid application development. The main advantage of XML is in its completely open structure, but the tools that build on XML provide the ability to construct iTAPS without having to reinvent the wheel. It also integrates nicely with ASP driven web forms.

c. Flash

Flash is a web technology developed by Macromedia Corporation (www.macromedia.com). It has a very robust and versatile feature set for displaying audio, video and web content. Flash provides a significant amount of customizability and has a rich feature set. It also runs on top of HTML/XML for maximum web browser usability and compatibility. The main desirability of Flash for iTAPS is in its ability to easily render graphics, which iTAPS needs for the display of the data diagrams. However, Flash has a lot more features than iTAPS needs as far as graphical features. In a sense, it is “overkill” for the iTAPS

feature-set. Another more important and significant problem with Flash technology is that it requires a client-side install and client side maintenance. These are issues that Second Fleet seeks to avoid to minimize client side configuration needs. These characteristics of Flash are significant negatives to implementing Flash technology in iTAPS.

d. Java

Java is a programming language and set of software tools developed by Sun Microsystems (www.sun.com). Java operates on the principle of cross-platform code independence. The user must install the Java virtual machine onto their client computers to enable the Java technology to function within web browsers and with Java applications. The software developer uses the Java Compiler to create bytecodes (applications or applets) that the Java virtual machine can execute for the particular platform the user is running. The principle idea behind Java is that by standardizing the software on a common code base, code that is written in Java can run on any machine that can run the Java virtual machine. Gosling and McGilton¹⁴ describe it as such:

Java technology is designed to support applications that will be deployed into heterogeneous network environments. In such environments, applications must be capable of executing on a variety of hardware architectures. Within this variety of hardware platforms, applications must execute atop a variety of operating systems and interoperate with multiple programming language interfaces. To accommodate the diversity of operating environments, the Java CompilerTM product generates *bytecodes*--an *architecture neutral* intermediate format designed to transport code efficiently to multiple hardware and software platforms. The interpreted nature of Java technology solves both the binary distribution problem and the version problem; the same Java programming language byte codes will run on any platform.

Architecture neutrality is just one part of a truly *portable* system. Java technology takes portability a stage further by being strict in its definition of the basic language. Java technology puts a stake in the ground and specifies the sizes of its basic data types and the behavior of its arithmetic operators. Your programs are the same

¹⁴ James Gosling and Henry McGilton; The Java Language Environment, A White Paper, May 1996

on every platform--there are no data type incompatibilities across hardware and software architectures.

The architecture-neutral and portable language platform of Java technology is known as the *Java virtual machine*. It's the specification of an abstract machine for which Java programming language compilers can generate code. Specific implementations of the Java virtual machine for specific hardware and software platforms then provide the concrete realization of the virtual machine. The Java virtual machine is based primarily on the POSIX interface specification--an industry-standard definition of a portable system interface. Implementing the Java virtual machine on new architectures is a relatively straightforward task as long as the target platform meets basic requirements such as support for multithreading.

Java is indeed a rich language with a rich feature set. Many applets are used daily by thousands of organizations and individuals on the Internet for applications from database servers to games. The problem with using Java for iTAPS is that it is strongly desired that no client side maintenance be required. While Java does has functionality that could be utilized to construct iTAPS, it would require Java knowledge on the user's part as well as IT administrative support to keep the Java virtual machine updated on all the clients. This is a significant problem as eventually the clients will be distributed among many deployed units and will not be managed within a single IT department.

e. *DAML (DARPA Agent Markup Language)*

DAML is an extension of XML that allows the creation of semantic structure within web documents. DAML is an evolving standard that uses XML and Resource Description Framework (RDF) to develop the schema or ontology within the metadata to allow intelligent agents to make inferences about the data they are processing. The DAML website¹⁵ describes the need for DAML as follows:

The World Wide Web (WWW) contains a large amount of information which is expanding at a rapid rate. Most of that information is currently being represented using the Hypertext

¹⁵ <http://www.daml.org/about.html>

Markup Language (HTML), which is designed to allow web developers to display information in a way that is accessible to humans for viewing via web browsers. While HTML allows us to visualize the information on the web, it doesn't provide much capability to describe the information in ways that facilitate the use of software programs to find or interpret it. The World Wide Web Consortium (W3C) has developed the Extensible Markup Language (XML) which allows information to be more accurately described using tags. As an example, the word Algol on a web site might represent a computer language, a star or an oceanographic research ship. The use of XML to provide metadata markup, such as Algol, makes the meaning of the work unambiguous. However, XML has a limited capability to describe the relationships (schemas or ontologies) with respect to objects. The use of ontologies provides a very powerful way to describe objects and their relationships to other objects. The DAML language is being developed as an extension to XML and the Resource Description Framework (RDF). The latest release of the language (DAML+OIL) provides a rich set of constructs with which to create ontologies and to markup information so that it is machine readable and understandable.

DAML appears to have the potential to be a very powerful tool but its use to iTAPS is limited at the moment. The problem is that there are no intelligent agents that analyze the iTAPS data. iTAPS is subjective human analysis aggregated into composite scores. Then the decision makers analyze these scores to affect future decisions. These scores are then analyzed by the decision makers to affect future choices. Future work on iTAPS could include incorporating intelligent agents to pre-analyze the data and make recommendations based on DAML compliant descriptions of ROE's (Rules of Engagement), but that is beyond the scope of this project. Another problem is that DAML is still very much in development, and is not truly standardized yet. As such, it would require too much specific management effort to keep the program working and upgraded.

f. ASP.NET

ASP.NET is Microsoft's newest extension of ASP within the .NET Framework. When examined closely, it is clearly evident it is also much more than an upgrade. While ASP supported scripting and database access support,

ASP.NET provides that capability and more, in a truly integrated format. It is important to understand the .NET Framework to fully understand ASP.NET.

The .NET Framework provides the basic system services that support ASP.NET. The .NET Framework consists of two main parts: the Common Language Runtime (CLR) and the .NET Framework class library¹⁶.

The common language runtime is supported by the Common Language Specification (CLS), which describes the data types that are supported by the runtime environment that are common to all the languages supported by .NET. The individual language compilers compile to an intermediate language called the Microsoft Intermediate Language (MSIL), which is then compiled by the runtime. This code is called managed code because it supports the CLS and is guaranteed to run on any .NET Framework compatible OS, similar to how Java code is designed to always run on any machine running the Java Virtual Machine.

The .Net Framework Class library is a set of base classes that developers can use to develop .NET compatible applications. A subset of these classes forms the basis for ASP.NET.

The main shift in ASP.NET is toward a completely Object-Oriented approach to coding applications (web applications in our case). Now developers can insert any type of code into ASP.NET web forms and have it be a true server-side program. The advantages of ASP.NET to iTAPS are numerous. First, it allows all code to run on the server with no client-side software except a web browser. This contributes to application stability and security. Second, it makes the development cycle shorter as many things that were manually coded in ASP are automatically supported in ASP.NET. Finally, ASP.NET allows central management and maintenance in a single location, the server. Specifically, ASP.NET improves on ASP in the following ways¹⁷:

¹⁶ G. Andrew Duthie, *Microsoft ASP.NET Step by Step*, Microsoft Press, 2002, p. 4

¹⁷ G. Andrew Duthie, *Microsoft ASP.NET Step by Step*, Microsoft Press, 2002, pp. 15-16

- **Web Forms.** This is the new programming model of ASP.NET. Web Forms combine the best of ASP with the ease of development and productivity of Visual Basic. You can drag controls onto a page and then write code to provide interactivity, call business objects, etc.
- **Server controls.** A major component of the Web Forms programming model, the ASP.NET server controls map approximately to HTML elements and provide server-side programmability.
- **Web Services.** This is a key part of ASP.NET that allows developers to make programmatic services available to other developers over the Internet (or local intranet).
- **Caching.** ASP.NET includes a new caching engine that will allow developers to improve performance of their applications by reducing the Web server and database server processing loads.
- **Configuration Improvements.** ASP.NET uses a new method for storing configuration information for web applications. Instead of having IIS store this information in a hard-to-access database, it is stored in XML based human and machine-readable configuration files.
- **State Management Improvements.** Overcomes the state management limitations of ASP and provides support for distributed session state across web servers, persisting state information in a SQL database, and providing state management without the use of cookies.
- **Security.** The security model in ASP.NET has improved authentication methods, code access security, and role-based authorization.
- **Improved Reliability.** ASP.NET contains new features aimed at improving the reliability of web applications, including proactive restarting of applications and automatic process recycling to resolve deadlock conditions and memory leaks.

Figure 3 illustrates where ASP.NET fits within the .NET Framework.

As can be seen, ASP.NET is part of an integrated solution between the CLS and the CLR.

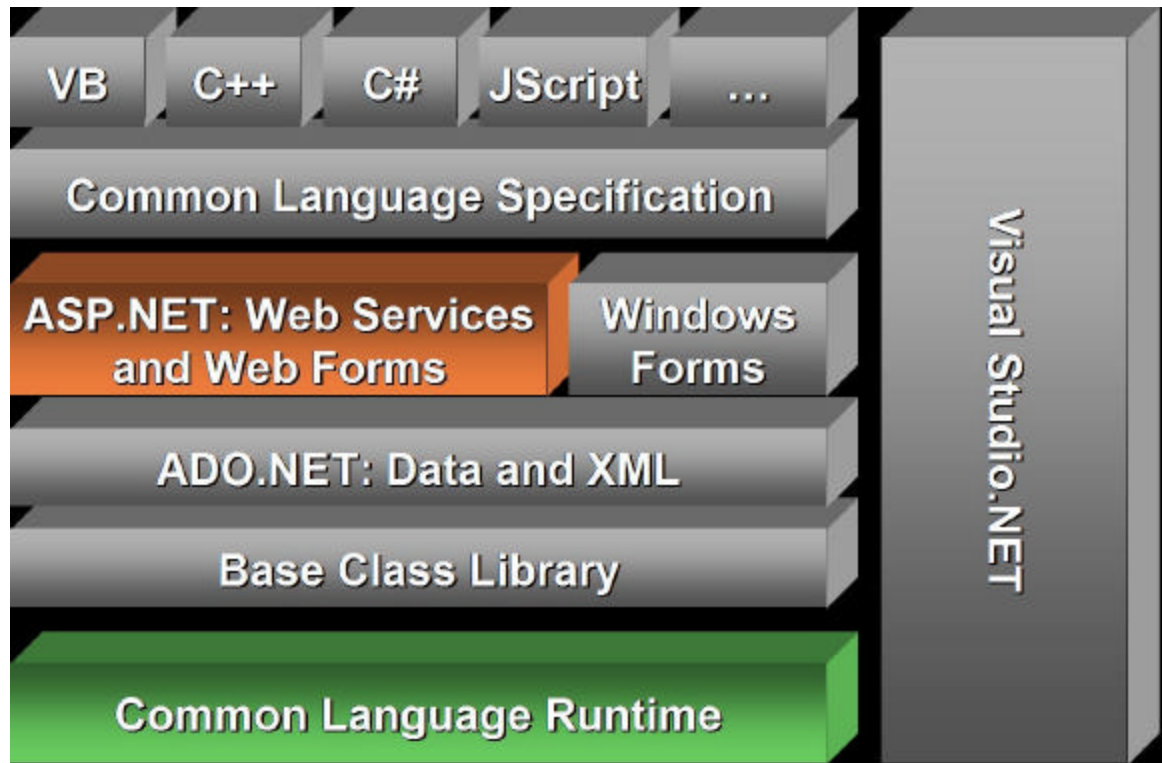


Figure 3. ASP.NET within the .NET Framework

ASP.NET provides distinct advantages for iTAPS. It takes the functionality of ASP and the flexibility of XML and fuses both into a single API. The advantages for iTAPS include:

- Reduced development time
- Minimal client side configuration
- Ability to implement graphics independent of Excel or PowerPoint

ASP.NET is also integrated with ADO.NET, the data access component of the .NET Framework. ADO.NET is an extension of ADO (Active Data Objects) that worked within ASP web documents for data access. ADO.NET is the interface for ASP.NET to data sources within the .NET Framework (see Figure 1).

D. TECHNOLOGY CONCLUSIONS

iTAPS is now defined as a DSS that follows the EIS model. We intend to support that model to the fullest extent possible to give the Commander, Second Fleet, a product that is intuitive, informative and flexible. The best choice for

database management appears to be SQL Server with web forms created in ASP.NET for access and control by the management users. SQL Server provides better tools and accessibility for a web-based enterprise. It also provides better integration with ASP.NET via ADO.NET. The best web development environment appears to be ASP.NET. With its integration with SQL Server and its feature set for web development, it appears ideally suited to developing iTAPS for web deployment.

III. THE DECISION SUPPORT SYSTEM

As discussed briefly in Chapter 2, iTAPS is a type of Decision Support System (DSS) known as an Executive Information System (EIS). There are additional DSS classifications that are useful in setting the boundaries of the iTAPS architecture and defining its roles and capabilities that we will explore in this chapter.

A. DSS DESIGN SYNTHESIS

iTAPS has the characteristics of several DSS technologies. Fundamentally it is an EIS, but that concept does not define it fully because iTAPS has significant *groupware* characteristics as well that influence the software design. Groupware is simply software that is specifically focused on issues related to collaborative processes among people¹⁸. While executive decision support defines its **purpose**, **structurally**, iTAPS needs to be a level 3 multiparticipant decision maker (MDM) system. More specifically it must be an Organization Decision Support System (ODSS) with the sub-characteristics of collaborative support technology known as a Collaborative Authoring System (CAS). Let's examine these terms and demonstrate why iTAPS must be structured as defined above..

1. Level 3 MDM

DeSanctis and Gallupe developed a classification hierarchy for Multiparticipant Decision Making Systems.¹⁹ Systems become more complex and functional as they increase in level. Level 1 systems are designed to only reduce communication barriers. Level 2 systems reduce uncertainty and noise during decision making. Level 3 systems regulate the decision processes. Using these definitions iTAPS qualifies as a Level 3 MDM. All of the characteristics of a level 3 MDM are listed in Table 2 and the characteristics pertinent to iTAPS are bolded.

¹⁸ M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 170

¹⁹ M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 168

<u>Participant Needs</u>	<u>System Features</u>
Enforcement of formal decision procedures	Automated procedure mechanisms
Increased clarity of options for decision procedures	Automated advisor for providing advice regarding various approaches
Structured and filtering of messages to adhere to rules	Structuring and filter agents
Development of deliberate governance rules	Rule set construction and inference mechanisms

Table 2. Level 3 MDM Characteristics (iTAPS in bold)²⁰

2. ODSS

An Organizational Decision Support System (ODSS) is a complex system of computer-based technologies that provide support for decision makers spanning the range of organization roles and functional levels and accommodates decision contexts that cut across organizational units²¹. iTAPS fits this structure precisely because it crosses many organizational roles, functional levels and boundaries. It is precisely this characteristic that gives the executive decision maker a broad view of the battlespace.

3. CAS

Collaborative Authoring Systems allow multiple participants to collaborate either synchronously or asynchronously in the time domain on the creation and retrieval of a common document or set of documents²². As iTAPS is an

²⁰ M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 168

²¹ J. F. George, "The Conceptualization and Development of Organizational Decision Support Systems.", *Journal of Management Information Systems* 8(3), 1991 pp.109-25

²² M. Marakas, *Decision Support Systems in the 21st Century*, Prentice Hall, 1999, p. 173

asynchronous tool that allows multiple authors to input data for synthesis and display for the decision maker, it clearly fits this structure.

It is important to understand these characteristics of iTAPS to have the proper background to build its structure. To recap formally: from the decision maker's perspective, iTAPS is an EIS, but from the users who support and input data to the system that provides those EIS functions, it is an MDM/ODSS/CAS. This is the framework from which we will define the complete DSS including inputs, outputs, view formatting and user interface design.

B. INPUTS

The DSS input context consists of administrators, input users, data and timing.

1. Administrators

Administrators are defined as users who enter structures, relationships and factors for computation within the iTAPS system. They are responsible for establishing and maintaining the relationships between graded elements and their weighting as needed by the executive decision maker as well as create and maintain the structures and contexts for the Input Users to enter data into the system.

2. Input Users

Input Users are defined as users who populate the iTAPS system with data. Input users are given a defined structure in which to input their data for consistency and ease of training. These users have their access restricted to their own data domains so as to prevent them from inadvertently damaging data in another input user's domain, thus entering erroneous data into the iTAPS system.

3. Data Description

Data will be entered and stored in the following contexts: Numerical grading, text notes, time-stamping, calculation and data structure. Numerical grades and notes are entered by the Input Users. Time-stamping and calculation

are done by the DSS. The Data structure is created and maintained by the Administrator Users.

4. Update Timing

The update timing is immediate upon Input User or Administrator User entry or change. The timing of the system is defined as asynchronous as different users can input data at arbitrary times.

C. OUTPUTS

The DSS output context consists of users, displays and data mining.

1. Output Users

The primary Output User is the executive decision maker; however, the decision maker's staff and the Input Users can derive benefit from viewing iTAPS by analyzing their respective categories and developing solutions to problems independent of, or to propose to, the executive decision maker. It is beneficial for all authorized command personnel to view the iTAPS displays to become more fully informed about the battle space and to view what the executive decision maker is seeing.

2. Desired Information Displays

The information displays are the heart of what iTAPS does. They provide the element grading information and generate the graphical radar diagrams that synthesize the underlying information for the decision maker. They also provide trend analysis and access to qualitative as well as quantitative information.

3. Archiving and Data Mining

All data is preserved in-perpetuity for trend analysis and analysis across contexts. In this way, it is envisioned that the actual decision support structure will be refined for each new context to more accurately measure what is important to the decision maker.

D. USER INTERFACE

We alluded in the previous section to the general format of the DSS displays. Now we will more fully analyze the current displays and what is expected and desired of the updated displays.

1. Current Structures

The current display structure consists of the following:

- A radar diagram with related area grades graphically represented along the spines of the “web”
- Descriptive labels at the end of each spine with hyperlink drill-down capability
- A descriptive title for each diagram
- A time stamp
- A link to information for source-level elements that includes
 - Owner
 - Comments
 - Timestamp
 - A navigation frame

2. Improvements

Some improvements envisioned for iTAPS are as follows:

- A hyperlink on each graded element to dynamically configurable trend and comment history
- A dynamically configurable trend analysis on each radar diagram by color
- The ability to add comments to composite elements
- The ability to view comments for composite elements
- The ability to dynamically data-mine across contexts for trend analysis
- The ability to recall and do analysis on any element from any context (data history preservation)

3. Alternatives

Several alternatives exist for view formatting. Most have been used before and have been found lacking by the decision makers when compared to the radar diagrams. One such format is the “stoplight” format (figure 4).

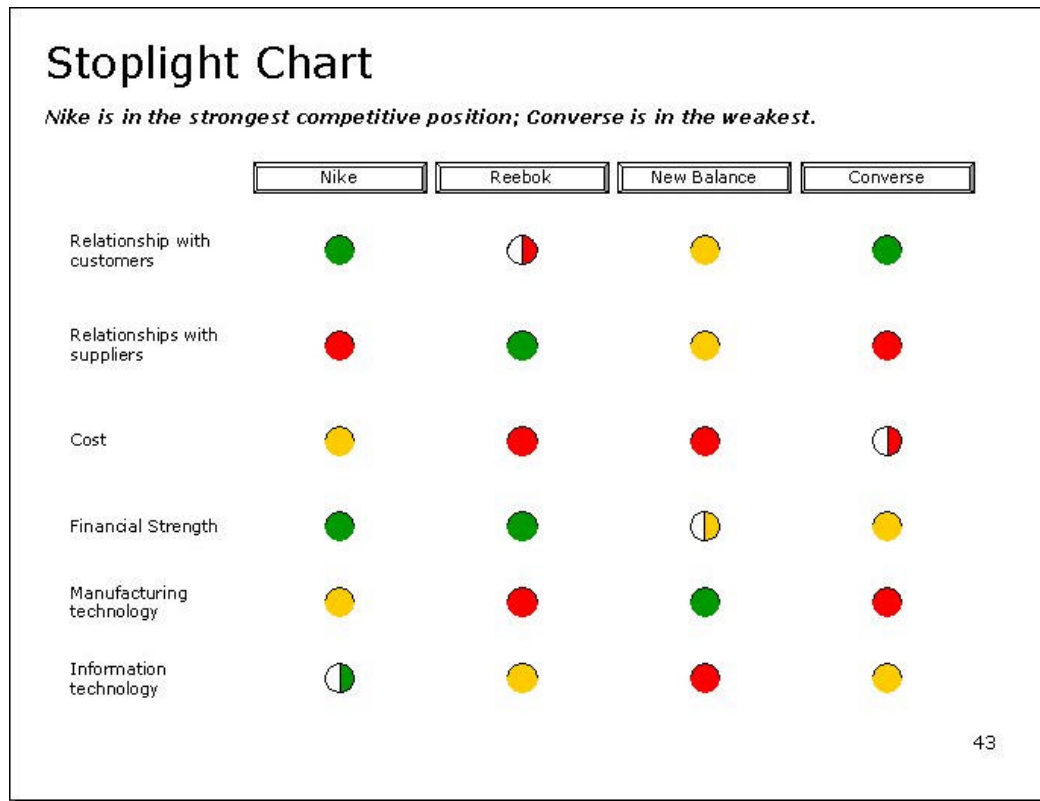


Figure 4. StopLight Chart Example

The problem with this format is its granularity. It has limited states. This tends to oversimplify the complex information that iTAPS is trying to synthesize. Another common format is the “color line” format (figure 5).



Figure 5. Color Line Chart Example

The problem here is too much granularity. In addition, trying to relate the color to a level of concern for that element is difficult. Questions such as “what does orange mean?” often arise. The concept of using iTAPS elements graded on a 0-10 scale is very accessible and understandable to most individuals. It is instantly familiar and once concern criteria are established, it is easily integrated into an alert triggering environment. We evaluate the current basic structure of a 0-10 grading system with a radar diagram display as both required and desired to adequately display the iTAPS data.

4. Flexibility

It is also desired that there be some flexibility in the iTAPS structure. Specifically, the most user configurable portion is the trend analysis area and the data-mining area. The user should be able to search on various categories and criteria without limitation so as to extract the data they desire.

D. USER INTERFACE DESIGN

There are several fundamental principles that are implemented throughout the entire design. They are as follows:

- Use of standard ASP.NET templates for fonts, TextBoxes, Labels and backgrounds
- Minimal use of memory intensive enhancements such as special fonts, colors or graphics so as to minimize bandwidth use on military networks.
- Use of a common theme and colorset throughout the web forms to present a consistent interface for the user.
- Minimizing bandwidth use by intelligent use of server-side code and compression technologies

Specific areas addressed are the data input pages, the administrative pages, the view pages and the user feedback mechanisms.

1. Data Input Pages

The data input pages are Windows Access Permission protected pages for Input Users to input their data. In keeping with the fundamental principles defined above and good net coding practice, the data input pages will use no background color and will use default-type fonts for display. They will also be developed with a common theme and recurring information will appear in the same area of the screen across contexts. Efficiency and brevity are key to this class of web form. The data input pages will provide pages for the following functions:

- View Owned Elements
- Select particular Owned Element for editing and/or entry from currently Active Phase and Operation
- Enter Grade Information

- Edit Grade (for user mistakes)

2. Administrative Pages

The administrative pages are password protected pages for administrator users to enter, update and alter the structure of the DSS to meet the decision maker's needs. They also provide functionality to enter, define and/or alter information within each database table. They will also conform to the user interface principles. The administrative pages will provide pages for the following functions:

- Create New Operation
- Delete Operation
- Edit Operation
- Create New Phase
- Delete Phase
- Edit Phase
- Create New Element
- Delete Element
- Edit Element
- Create New GraderTitle Entry
- Delete GraderTitle
- Edit GraderTitle
- Create New Grader Entry
- Delete Grader Entry
- Edit Grader Entry
- Create New ContactPerson Entry
- Delete ContactPerson Entry
- Edit ContactPerson Entry
- View all Operations
- View all Phases
- View all Elements
- View Operations
- View Phases of selected Operation

- View Elements of Selected Phase and Operation
- View and select from list of all Operations
- View and select from list of Phases within an Operation
- View and select an Element for editing or insertion from a list within the selected Phase
- Edit Grade

3. View Pages

The view pages are accessible to all authorized users and are the output of the DSS. There must be graphical context to properly display the radar diagrams but bandwidth considerations will still be made. They will conform to the user interface principles above to the greatest extent possible. The view pages will provide the following functions:

- View Radar diagrams of Active Phase and Operation with drill down capability
- View Particular Element grade trends over time

4. User Feedback Mechanisms

There will be two types of user feedback mechanisms. One is automated through the use of the time-stamping feature of SQL Server. The other is the manual email hyperlink structure utilized throughout web-based technologies. This will enable manual, user configurable feedback on DSS operation and corrections to a user's input. Contact information will be provided on screen for each Input User in each grading area.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. METHODOLOGY

A. EVALUATE CURRENT SOFTWARE CAPABILITIES

1. Findings from Research

The basic findings of the research indicated that there were numerous avenues that could be taken to develop iTAPS. As seen in Table 3, there is no doubt that a robust, quality product could be developed using ASP, Java, or Macromedia Flash. In the end, we were looking for a product that could meet or exceed all our hard and fast requirements, as well as the features that were “highly desirable”.

<u>Requirements</u>	ASP.NET	ASP	Java	MM Flash
Radar Diagrams	X	X	X	X
Online Data Input	X	X	X	X
Drill Down Capablity	X	X	X	X
<u>Desired Features</u>				
No Client Side Code Required	X			
No Cookies Required	X			
Integrated Security	X	X		

Table 3. Development Product Comparison for iTAPS Requirements

The one feature that removed most of the possibilities was the requirement of not having to install client side code. This was not a hard and fast requirement because of the many powerful technologies which implement this type of architecture but removing client side code is unquestionably a “highly desirable” trait. The problem with the use of client side code is that it requires maintenance and management. Ordinarily, this extra maintenance would be trivial to an IT administration staff because browsers incorporate automated updated routines which automatically download and install the needed code. iTAPS however, will not be running on the internet, but rather the Secure Internet Protocol Routed Network (SIPRNET). This means that all the automated tools to load client-side updates automatically will not function. The Second Fleet IT staff maintains hundreds of terminals on their command ship the USS Mount Whitney. Having the extra task of properly configuring and maintaining each machine with client-side code such as the Java Virtual Machine or Macromedia Flash is an undue burden to the Second Fleet IT staff.

Another problem with required client side code is that it could compromise the confidence users have in the iTAPS system. While iTAPS can and will be used for real world operations, the bulk of iTAPS use will be during Joint Force Exercises (JTFEXs). During these exercises most of the users will be personnel temporarily assigned to COMSECONDFLT and completely unfamiliar with iTAPS. Any stumbling blocks placed in the way of users will undoubtedly reduce user confidence and in turn their support of iTAPS.

2. What Was Chosen and Why

We chose the .Net Framework implementing ASP.NET and ADO.NET to develop iTAPS. The .Net framework was the one product that removed the requirement of client side code, while still providing all the required functionality to make iTAPS a robust decision support system.

The .Net framework is indeed a paradigm shift for developing software on Microsoft platforms and the list of features it provides is noteworthy. However

there are a few principal features that set the .Net framework apart from the other possible solutions.

The first benefit is that the .Net framework is that it is an entirely Microsoft solution. This was a preference expressed by the Second Fleet Staff to stay in line with the current Navy doctrine of standardizing on Microsoft products. From an administration point of view it simplifies administration of the various servers needed to develop a DSS like iTAPS. From a development point of view, a complete Microsoft solution streamlines integrated security and easily provides Software System Integration and Complete System Specification Support.

Another benefit of .Net is gained from using ASP.NET. ASP.NET provides a sophisticated capability to maintain session state without the use of cookies. One of the challenges of using scripts to program web pages in the past was that client-side cookies were the only way to maintain session state when a user clicked from page to page. Each web page is treated as a separate and distinct program. Storing a cookie on the client enabled data to be passed from page to page to maintain the session state. ASP.NET now provides a feature that allows session variables to be used in the server-side code to maintain various elements of the session state without the need for storing this information on a client-side cookie. Not requiring cookies will also allow tighter security configurations for client browsers, as well as removing the problem of “the system isn’t working” when cookies are required but have been turned off on the client browser.

A third benefit of using the .Net framework is that by using Visual Studio.NET developers can dramatically reduce the development time of iTAPS. The Visual Studio.NET Integrated Development Environment (IDE) incorporates the Visual Basic method of programming for all its available programming languages. Objects can simply be dragged onto the page and the requisite code needed to make them work is automatically generated in the background. In addition, because the .Net framework uses the common language runtime (CLR), one language can be programmed against and debugged in any other language

supported Visual Studio .NET. This will provide a great deal of flexibility for future versions and maintenance of the iTAPS code.

The final benefit of using the .Net framework is that it has an eye for the future. The underlying framework of .Net capitalizes greatly on Extensible Markup Language (XML). Because of this, any data or processes generated from iTAPS can easily be connected and shared with other applications. XML is viewed as a technology that will enable the internet of the future. While we might not know what the future has in store for the Internet, we do know that by building iTAPS using XML, it should easily integrate into the software of the future.

B. DEVELOP WEB-BASED SOFTWARE SOLUTION

1. Architecture for Solution Chosen

The architecture we chose for iTAPS is a classic 3 tiered architecture featuring web clients as the presentation tier (any Microsoft Internet Explorer browser client version 5.5 or newer), the application server for the business tier (Windows 2000 running IIS), and a database for the data tier (SQL Server). See Figure 6 for reference. This architecture simplifies development as each portion of the DSS is broken down into smaller parts that can easily be developed. A 3-tiered architecture also provides excellent scalability which was one of the requirements for iTAPS. Lastly, having the application on several machines spreads the processing load so that one machine does not become overwhelmed with requests.

iTAPS Architecture

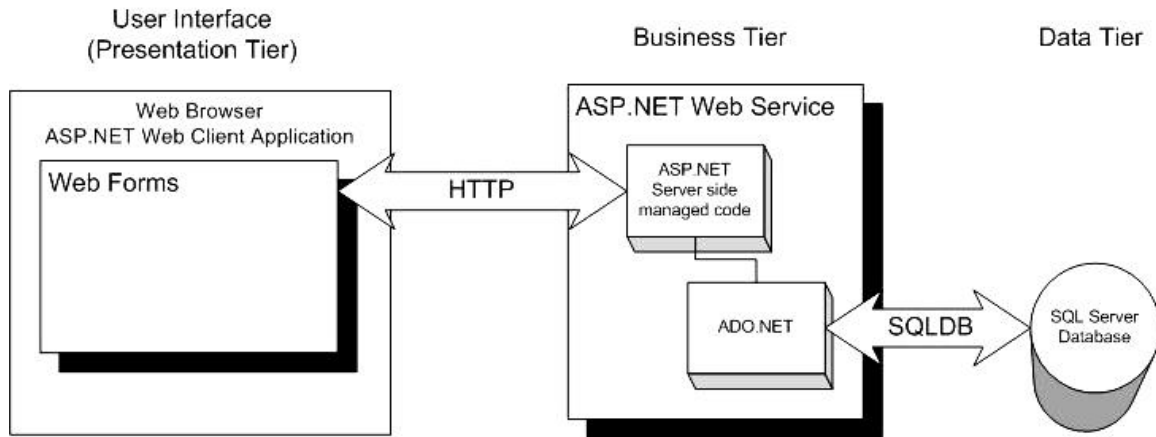


Figure 6. iTAPS 3-tiered Architecture

2. System Design

a. Database

(1) Design. The design of the iTAPS database was an extremely important part of developing the DSS. A poor design can place many limiting factors on what can be displayed and what can be stored for future reference. The iTAPS database was designed with several goals in mind. We wanted to capture as much data as possible to provide commanders with the flexibility of presenting this data in many different views, with any desired level of granularity. The chosen design would also allow for identifying short term trends for enhanced immediate decision making.

As with all software development, we had to scope this project for the time available. Therefore, we could not incorporate all the features we wanted in the first version. However, with an eye towards future versions, we included some capabilities in the database that are not implemented in our iTAPS code. By designing the database in such a fashion, future versions of the iTAPS code can easily incorporate these features without having to completely reengineer the system. One envisioned future feature is the capability to set "trip wires" whereby some type of action (emails sent, flags on the charts displayed) is

initiated when certain criteria in the data are met. The other forward looking feature is that every operation is archived in the database and is designed to facilitate data mining to examine long term trends once enough data is collected even though there is no data mining application code.

(2) Linking to Web Pages. To connect the database to the web pages for iTAPS the .NET environment uses a SQL server data connection. This establishes a connection from the program running behind the web page to the database. When data is needed a SQL data adapter is used to actually retrieve the data from the data base. The SQL data adapters accomplish this by using SQL statements or stored procedures on the database. Stored procedures are simply saved sets of SQL statements that can be reused over and over. For our purposes stored procedures generally worked best because they also allow you to pass values to variables that are then used in the stored procedure. This provides the flexibility needed to facilitate the interaction with the database. For example, the stored procedure below will delete a Phase based on a PhaseID. By using the variable Original_PhaseID (designated by the @ sign), you can specify exactly, dynamically in code, which Phase will be deleted.

```
CREATE PROCEDURE dbo.PhaseDeleteCommand
(
    @Original_PhaseID uniqueidentifier
)
AS
    SET NOCOUNT OFF;
    DELETE FROM PHASE WHERE (PhaseID = @Original_PhaseID)
GO
```

b. Web Pages

(1) Creation of How Pages Look. Web pages were designed with simplicity in mind. The sole purpose of these pages is to input, synthesize, and display data. Nothing was placed on a page that did not contribute to these tasks. Colors were limited to black, white and blue unless there was significance in the color on a graph. For example, charts displaying "Red Side" (enemy) information are displayed using red lines.

(2) Programming Behind the Web Pages. The iTAPS web pages were designed using the web forms tools in Visual Studio. These tools use the Visual Basic method of rapid development where a user can simply drag items on the page and all the code necessary to facilitate that object on the web page is automatically generated. The power of this tool can not be overstated. This feature works not only for the objects displayed on the web pages such as buttons and drops lists, but also non-displayed objects such as data adapters, and datasets. This significantly increases the speed of development.

Because iTAPS is a 3 tiered application there are 3 main parts to the web page creation of an iTAPS page – the logic, how it is displayed, and interaction with the database. ASP.Net makes use of any of the Visual Studio Languages (in our case we used C#) to provide the logic behind the pages which are then displayed on the client using HTML. ADO.Net uses XML and SQL commands to facilitate interaction with the data base. It is easy to see that there are many requirements and many technologies being used to develop each page. However, that is why the Visual Studio click and drag object creation is so powerful.

To demonstrate we can walk through creating a page. If a page is needed to display some data from the database it will need a data connection to establish connectivity to the database; a data adapter to define exactly what queries will be run against the data base; a dataset to hold the query results in memory; and a display object such as a data grid to display the data on the page. Furthermore, the webpage will need HTML to display its contents. Each one of the objects can be created by dragging an icon onto the page and modified by right-clicking the object and changing its properties. Some code will be needed to bring everything together, but a developer need only type 3 lines of code to get the page to display the retrieved data. This is a fraction of what would otherwise be required to create the same page typing all the code manually.

(3) Generating Charts. Dynamically generating all the different shaped radar charts in iTAPS is one of the most complex items that are displayed on the iTAPS web pages. To simplify this process, the charts in iTAPS were created with a software package called ChartFX created by the company SoftwareFX. ChartFX simply provides additional classes created by SoftwareFX that are added onto the .Net framework classes giving you chart objects that can be easily created and tied to data using the same Visual Basic style of drag and drop creation. By using ChartFX we save ourselves the need to generate the immense amount of code needed to manually create the iTAPS charts. ChartFX also has a built in drill down feature that easily facilitated meeting the requirement of having clickable links on the charts to bring up charts with the subordinate data. These features were all accomplished while still meeting our goal of not requiring any client side code.

c. Integration with the Application Server

(1) Integration of Security Measures. Our goal was to keep the security for iTAPS strong but unobtrusive. One of the benefits of standardizing on Microsoft products is that all of their products come with Windows Integrated Security. Each product lets you seamlessly integrate control access with enough level of granularity to ensure that only the people who need access to particular data can get to it, and people who should not see anything will be kept out.

For the most part limitations are controlled by groups. One example is how particular directories used by Microsoft Internet Information Server (IIS) can have access limited to a particular group to ensure that only that group has access. However sometimes you do need a more specific limitation on an object. For example, a grade can only be entered in by the one specific person assigned to enter that grade. Programs written with Visual Studio.NET allow you limit the use of the grade object by associating it with one specific account.

Defining groups to limit access is simplified by using Windows user groups on the Windows 2000 server. Once groups are established

their limitations are implemented by simply clicking a check box “use Windows Integrated Security” in the application you are using (IIS, SQL Server, Visual Studio). This then allows you to define exactly what limitations you want to implement.

For iTAPS we have established 3 groups that define what a user can see and do.

- a) Administrative Users have access to everything. They will actually create the data structure for operations and facilitate the DSS.
- b) Input Users are the graders. They will be able to look at charts as well as enter grades only for elements that they have been assigned to grade.
- c) View Users only have access to view the charts created by iTAPS. By definition view users include everyone with access to the Windows domain where iTAPS is deployed.

(2) User Authentication. Windows Integrated Security is the basis of Windows Authentication Protocol (WAP). As discussed above access is controlled by access to the Windows domain and membership to particular Windows user groups. Therefore authentication for access to iTAPS is all handled by WAP. This is fortuitous because a user will not have to remember a new username and password. All the user needs to do is log on to the Windows domain and his user name and password provides all the authentication credentials necessary to access iTAPS.

4. System Development

a. Development Methodology

Because the scale of the project was relatively small and we had to respect our time constraints, we chose the Extreme Programming Methodology to develop iTAPS. Following the methodology of Extreme Programming, we took the established systems requirements, developed minimal pseudo code, and then began programming. For the most part this methodology was very

successful. We could rapidly generate quality code for the web pages. Occasionally (as is the case with Extreme Programming), code for a particular page had to be completely scrapped and the page had to be completely rebuilt. This fact did not have significant consequences however, because the code for web pages is relatively compact and inherently modular. On the rare occasion when we did scrap code and start with a fresh approach, it caused little to no impact to the established pages, and the project timeline.

b. Test Lab

iTAPS was developed in a test lab consisting of a server with two clients. The server was running Microsoft Windows 2000 with IS, SQL Server 2002, and the .NET framework. As depicted in figure 6 it is optimal to have servers separated on the own individual machines, but for development purposes there is was no need to spread load the processing, as the client tasking on the server was minimal. Because we were not running our own Windows Domain Server, the Windows 2000 server was also added to the Naval Post Graduate School “NPGS” Windows domain to provide sample Windows domain users.

The two client machines were running Windows XP. They were used to actually code the application, as well as to test the application by functioning as a client with an authenticated user. Visual Studio.NET was used to construct the code, and Internet Explorer 6.0 was used to function as a client to test the code on the server. It is important to note that the operating system of iTAPS client is not overly important. Internet Explorer 5.5 or 6.0 is a requirement, but either one will work with a machine running a Windows operating systems based off of Windows NT (NT 4.0, Windows 2000, and Windows XP).

c. Testing Methodology

Because iTAPS consists of inherently modular, loosely coupled programs that generate its pages, we chose to focus on “black box” testing for our testing methodology. The “black box” method of testing is described as examining individual pieces a system and testing them individually for there functionality. While an integrated testing plan (“white box testing”) is always complementary to “black box” testing, time constraints limited the entire amount

of testing that we could accomplish. In the end, we did not feel that we would be seriously compromising the quality of the system because the programs running the web pages are so loosely coupled. There is a small degree of interaction by way of session variables for the individual programs running each web page. However, after you conduct thorough “black box” testing of the code for a page, errors involving session variables will be readily apparent once the pages are integrated.

Another method we employed to debug our code was to grant access of our beta versions to our users. Our main goal was to obtain feedback from our customers to improve the application, but this provided the added benefit of getting more eyes on the application to find bugs. This type of input was extremely useful because users often created scenarios that we had not thought of for the system. In the end these situations either had to be addressed in the code or prohibited through business rules.

d. Documentation

When iTAPS was completed, a thorough User Manual (Appendix 1) was created. The User Manual is broken into two parts - how to install the application and how to actually use it. In both sections, step by step walkthroughs are included with multiple screen shots to explain in detail how to make iTAPS work. The application itself also includes “tool tips” where appropriate. By simply “mousing over” a button you can get a short description on what that button will do.

5. System Delivery

Once iTAPS was completed Visual Studio was used to create an installation file that could be used to install iTAPS to a new server. This file and all the other requisite files to make iTAPS functions were all burned to a CD to facilitate a complete installation (for installation details see Appendix 1 – iTAPS User Manual). The final package was delivered to the customer Commander Second Fleet on 17 September 2002 aboard the Second Fleet Command Ship the USS Mount Whitney.

Aboard the Mount Whitney, iTAPS was successfully installed by Second Fleet IT personnel on a Windows 2000 Advance Server running IIS and SQL server. This server was connected to their experimental LAN. Their experimental LAN's domain controller was used to establish the Windows groups to facilitate the integrated Windows security. Once the installation was complete the system was tested with a Windows NT 4.0 client running Internet Explorer 5.5.

After a successful installation and test, we conducted a brief for the principle administrators of iTAPS, the staff officers of the J35 (operations). iTAPS was well received, with a high level of interest from the staff and even some recommendations for fine tuning the interface. Their recommended changes had to do with slight modifications to charts displaying data and were all completed on the spot. When the changes were complete, the application was recompiled and a new CD was burned to capture the changes. iTAPS will receive its first actual use for an exercise during JTFEX 1-03 during the month of October 2002.

V. FINAL RESULTS

A. SYSTEM CAPABILITIES

The Integrated Theater Assessment Profiling System (iTAPS) version 1.0.2 (final version for this thesis) met all the goals that were set at the beginning of the project. Specifically, all the functionality of the original TAPS is now part of the web-based iTAPS. Every feature of the new system is easily accessed by using a web browser. This includes developing a hierarchy of elements to grade an operation, inputting grades for the various elements, and viewing the graphs with the processed information. iTAPS gives the Commander, Second Fleet a near-real-time, web-based decision support system that provides an accurate profile of a theater during an operation or exercise. Information input to or viewed in the system can be provided by any one who has access to the Second Fleet network based on Windows Access Permission assigned by the ITAPS Administrators. The information is presented in a clear and concise manner, with the ability to drill down to various levels of granularity (see Figure 7).

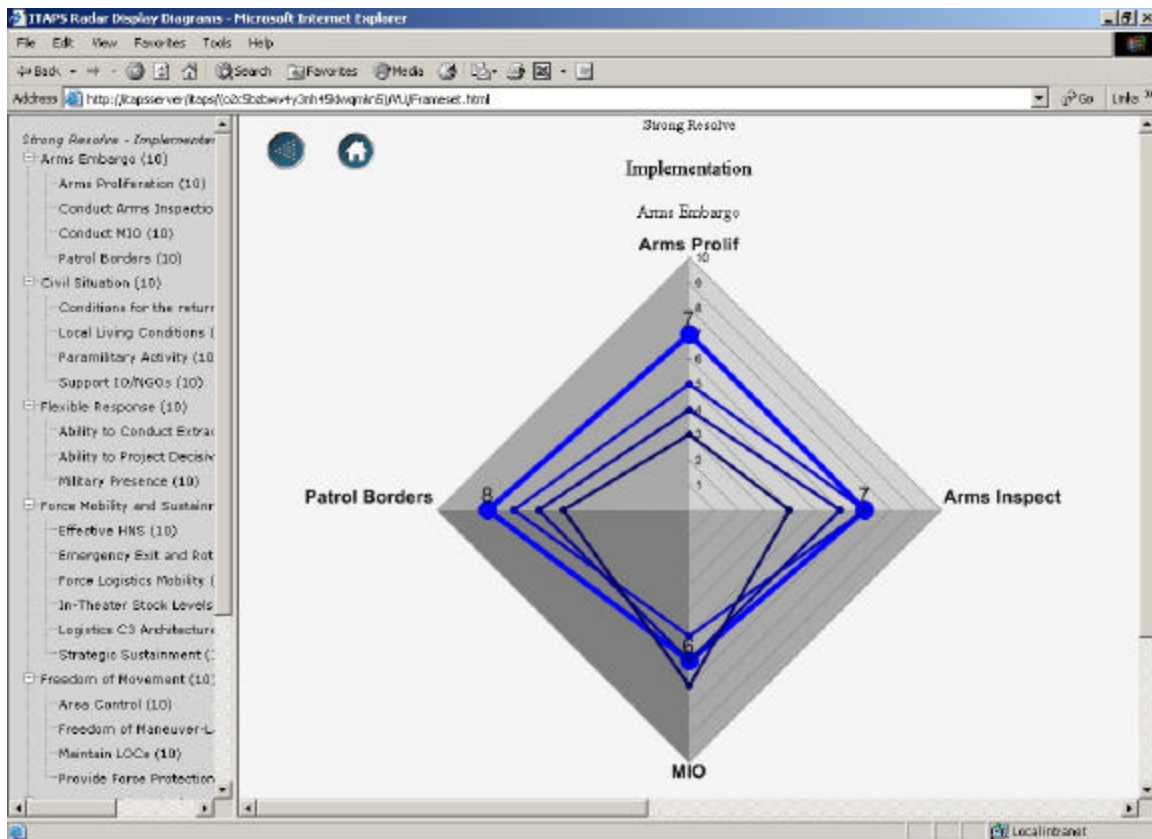


Figure 7. iTAPS Screen Shot

In addition to the primary goal of recreating the original TAPS to a near-real-time, web based system, several additional features were added to iTAPS.

1. Ability to Copy an Operation.

Second Fleet conducts JTFEX training for each Carrier Battle Group and Amphibious Ready Group deploying from the east coast. Because of the number of exercises, scenarios for JTFEXs are often reused. Therefore, many of JTFEXs are very similar in nature. We included a feature that allows the iTAPS administrator to copy the entire structure of an exercise (operation) minus the grades that have been entered. This can save the administrators a great deal of time because they can now develop a library of exercises that can be pulled up and modified in iTAPS to meet the specific changes of an exercise or operation.

2. Ability to Establish Side Color

During our research, some of the Second Fleet staff officers asked about generating TAPS charts for the enemy as well as friendly forces. In the new iTAPS we have included a side color feature. When an exercise is created, a side color is picked to determine what units will be graded. Generally this will be the blue (friendly) side and all of the charts will be drawn using blue lines. However, if the staff would like to assess the red (enemy side) as well, we have included a feature to copy the structure of an exercise (operation) with empty grades and a new side color. As soon as new grades are entered, charts using the new color can be generated as well.

3. Variable History Feature

In the old version of TAPS, grades were manually calculated twice a day and no grade history was retained. Only the current and previous day's grades were displayed on the charts. In the new version of iTAPS, grades will be updated continuously. In addition to the most current grade, 3 historical grade series are also plotted on the graph (see figure 7). These series can be set to show intervals at 8, 12 and 24 hours. This will give the commander much more useful information as he can easily see trends on the charts from up to 4 days in the past.

4. Grade Element Detail

In the old version of TAPS when you drilled down to the bottom of the grade hierarchy, a page presented what section was responsible for entering that grade, what the grade was, a date time stamp, and any comments the grader had (see figure 8).

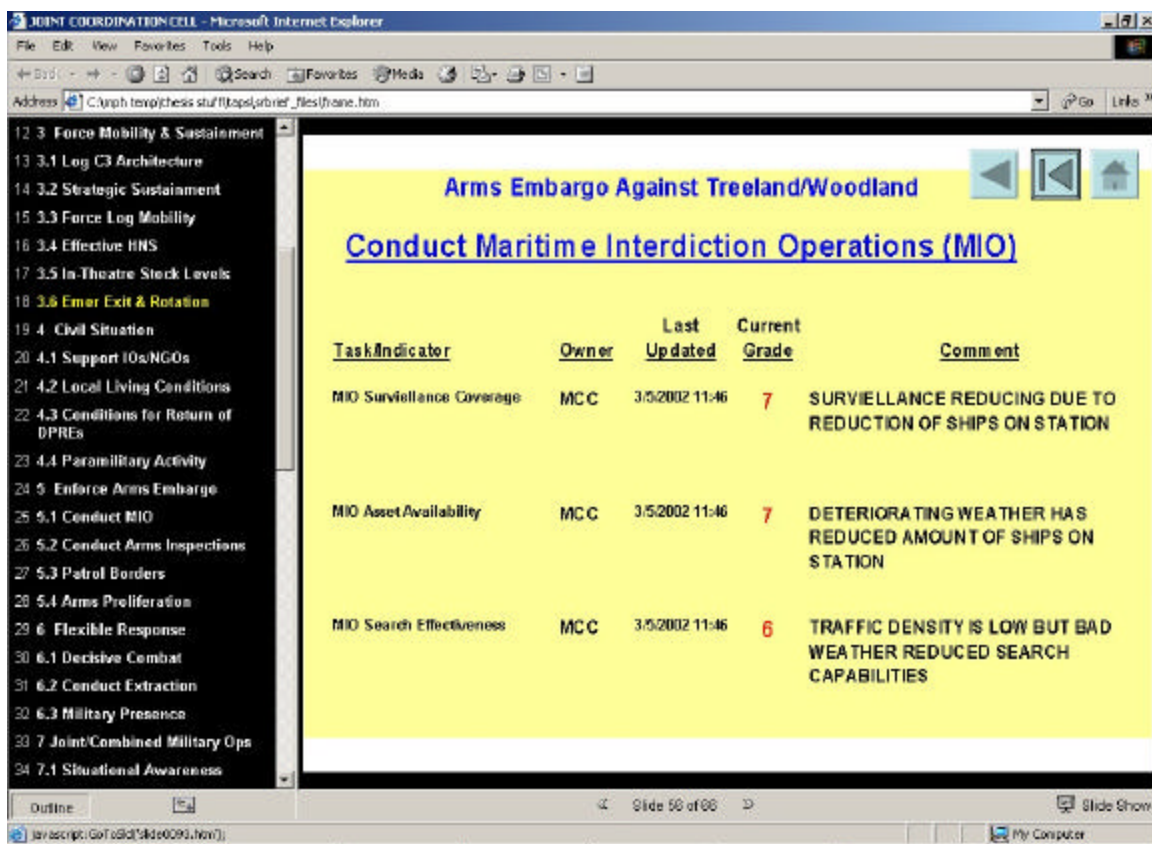


Figure 8. TAPS Grade Detail

The new iTaps now how has a more robust display. The page shows the grade; the date time stamp; the graders comments; what their name is and where they work; a clickable email address; an optional hyperlink to information relevant to the grade; a small chart showing the trends of the grade; and a data grid with all of that information for the previous grades (see figure 9).

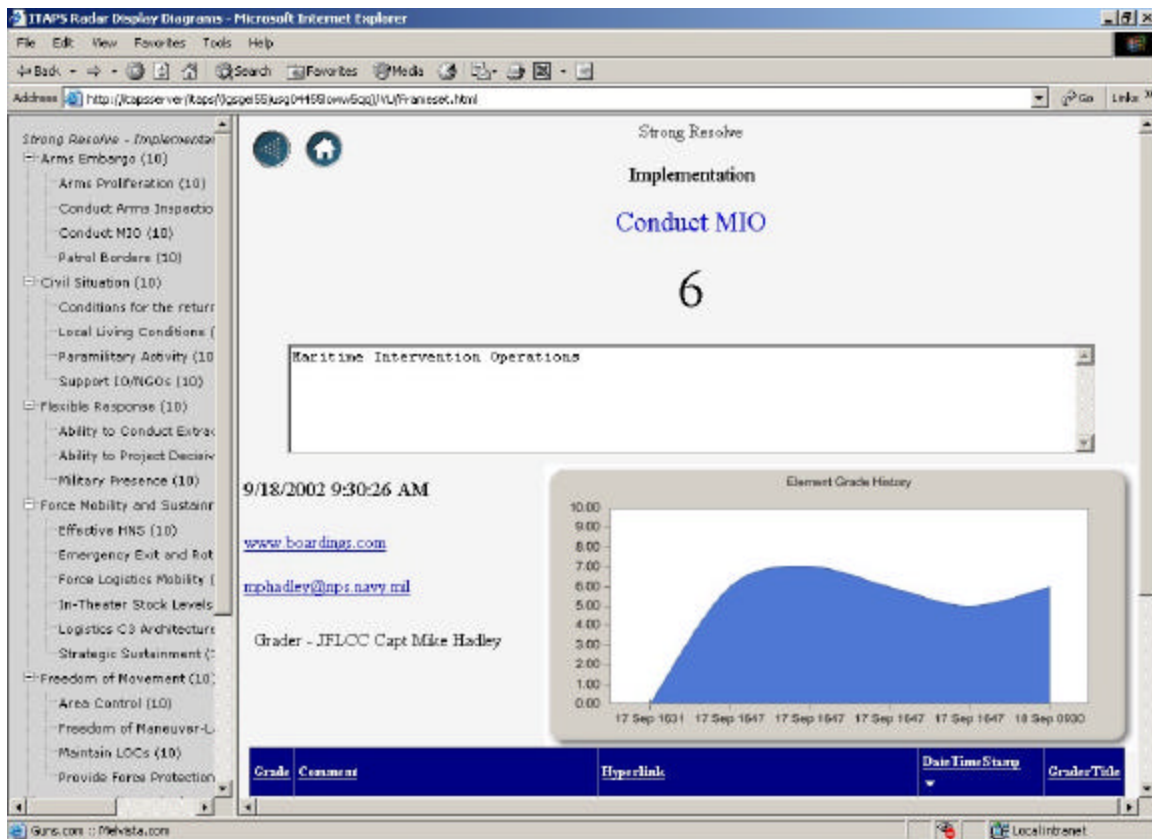


Figure 9. PS Grade Detail

B. SYSTEM LIMITATIONS

Despite the many advances made over the original TAPS program, there are still limitations to the new system.

1. Single Source Software Development.

Using the .Net framework gives a developer immense capability. In fact, it would be a fair statement to say that this system could not have been designed in the timeframe given without using the .Net framework and Microsoft Visual Studio. However, with this immense capability you do place limitations on the deployment of your application. Using the .Net framework means that you are strictly dealing with Microsoft products, to include the client side browser. For the most part this is no problem because the Navy is standardizing on Microsoft products. However, the Navy does use other Operating Systems for some of its applications.

One of the greatest challenges at Second Fleet is that it is also a NATO command and therefore there are multiple levels of classification of information. In the past this fact was dealt with by simply running multiple networks to keep the different levels of classification separate. Obviously, this provides a huge additional burden on everyone who manages and uses these systems. Instead of running one network to support the staff, the IT department must buy and run 3 of everything (servers, clients, hubs, cables etc.) The emerging solution to this problem is to develop a network that can function in a secure manner with multiple levels of access.

While we were installing iTAPS, another team was simultaneously installing just such a system – the Coalition Data Server (CODS). CODS is an experimental system used to manage multiple levels of access on the same network. That system was based on computers running the Sun Trusted Solaris operating system. This would not categorically preclude iTAPS from being integrated with CODS but there are some issues to address. Trusted Solaris was chosen because of its strong security capabilities. However, any time you increase the number and types of connections to the server (as would be necessary to integrate iTAPS), it increases the potential for a vulnerability. Security more often than not is a compromise on how much exposure you are willing to accept for your system. Therefore, while on the surface it appears that iTAPS could be integrated in with a system such as CODS, there are definitely issues that need to be addressed to integrate the Microsoft products while still maintaining the integrity of the system.

2. Scalable Output

Everything that is displayed on the iTAPS screens was designed to fit on a screen with a 1024X768 resolution. Generally this should not create any problems, but occasionally some people like to adjust screen resolution or the font size on their browser. In some cases that can cause some unexpected and undesirable results.

Another issue related to scalability is that as was noted above in capabilities, iTAPS can track and graph grades for different participants in an operation (red side, blue side etc.). The problem is that the graphs are not scalable as well. Therefore if you want to present two views of graphs generated for different side colors side by side, they won't fit on the same screen. It is easy to shift from screen to screen but it will not provide the same effect as having the graphs side by side.

3. No User-Friendly Grader Override

iTAPS was designed to maintain complete integrity of the data that is input to the system. Therefore the only the person that can enter a grade for an element is the person with the account that is assigned to that element as a grader. The only problem with this scenario is that past experience has shown that on occasion, for various reasons, a person might be unavailable to input their grades.

Currently the only solution to this problem is for an administrator to use the administrator tools and change the account linked to the specific grade, have the person (perhaps themselves) with that account input the grades and then change the grader assignment back to the original person.

This Second Fleet staff does not expect this to be a common occurrence, but obviously a more elegant solution should be developed to address this situation. Future versions of iTAPS might base these fields on a unique grader-type group for each grade, and allow multiple users to be placed into the group.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. CONCLUSION

The creation of iTAPS was a response to a direct request from Second Fleet to upgrade their current capability for theater assessment. The first conclusion that can be made regarding iTAPS is that it has definitely met all of the stated requirements in the original request put forth from Second Fleet.

All of the original capability in TAPS has been included in iTAPS but is now accessed with a web interface. In addition, beyond the new web interface, several new features have been included to improve ease of use and increase the benefits derived from using the system.

Much of the success of iTAPS can be directly traced to the decision to use the .NET framework and Visual Studio.NET. It would be foolish to make the statement that this combination is the “best” developmental platform because every scenario is different. However, in our case, we can say with confidence that it was in fact the best platform for us to use.

The .Net framework and Visual Studio.NET offers some very powerful features that allow a developer to rapidly generate intricate web based applications. The main features that helped us were the ability to include entire programs in the ASP.NET pages; no requirement for client side code or cookies; and the drag and drop development capabilities of Visual Studio.NET.

- **ASP.NET.** In the past ASP script or Java script had limitations as to what type of coding could be included on a page. With ASP.NET we could easily code advanced logic using traditional programming techniques and still include all of the code in our web pages. This is undeniably a paradigm shift for web page development.
- **Client Side Code.** The ability to not require client side code or cookies was not absolutely necessary for the development of iTAPs. However, the benefits of this capability could not be ignored. Java from Sun is one of Microsoft's main competitors. There are many things that can be said to praise Java, but it will always require the Java Virtual Machine to be installed on every client. When presented with a similarly capable developmental platform that does not have this extra requirement, it was in our

best interest, in accordance with our customers requirements, not to use Java.

- **Drag and Drop Development.** The drag and drop development environment of Visual Studio.NET is an extremely powerful tool. That environment not only reduced development time, it also greatly simplified an otherwise complex process of integrating many different technologies. Because of this environment we were afforded the ability to add many additional features to iTAPS and provide the ability to interface with other applications through XML.

By selecting the .NET framework and Visual Studio.NET as our development environment, we did limit ourselves in some ways. But in this instance the decision was not entirely limiting. For our purposes Microsoft could provide everything we needed to rapidly develop a product that met all our requirements and more.

In the end iTAPS not only provides the Commander, Second Fleet with a solid theater assessment tool, it provides a product that will lay the foundation for theater assessment in the future, with scalability, adaptability, and the ability to integrate with systems of the future.

VII. RECOMMENDATIONS FOR FURTHER RESEARCH

As with any software project, time will always limit the amount of functionality that can be coded into an application. In addition to the limitations discussed above, the following items are some of the many things that we think could be developed to improve iTAPS in the future.

A. PRESENTATION OF DATA WITH RADAR CHARTS

The radar or spiders charts (see figure 7) used to display data in iTAPS effectively convey the information to the user. However, the details of maximizing the effectiveness of creating charts using the radar format were never researched. Undoubtedly studies have been conducted to determine what the most effective line size, colors, etc. are. Researching this information and modifying iTAPS to incorporate the findings could greatly enhance the data presentation elements of iTAPS.

B. ADD TRIP WIRE CAPABILITY

iTAPS was created with the ability to turn on a trip wire capability. The idea is that when certain criteria are met in the data, the system could respond by informing the staff in various ways that an important event had transpired. The response could possibly be automatically generated emails, or changes on the output displays. More research and input from the Second Fleet staff could help develop ideas, as to exactly what they would desire to happen when a trip wire value has been reached.

C. ADD DATA MINING FEATURES

One of the best things about deploying an enterprise level database is that once it is used for a while, there is a wealth of data that can be “data mined” to provide new insights to the users of that database. While iTAPS was in fact designed to facilitate data mining, it currently does not have this feature. Research could be done to determine exactly what type of historical information would be useful to the Second Fleet. Once that information is determined, additional features could be added to the Administrator options to allow data to be extracted, processed, and presented in a useful form. This way the Second

Fleet Staff could select the various historical operations and exercises, analyze them, and then use that information to improve decision making or forecast future events.

D. IMPROVE DRILLDOWN CODING

The drilldown code of iTAPS is fully functional, but it does incorporate some code that would be considered poor programming practice. The issue is that when a link is clicked on to drill down to the next level of information, there is a problem with part of the key being passed from page to page. That key is what is used by the SQL query to retrieve data for the next page. The actual key fields being used to execute that query are the Operation Unique Identifier, the Phase Unique Identifier and the Element Title (see Appendix B for the Data Base schema). The problem is that it is poor programming to use actual data (the Element Title) rather than a unique identifier to execute queries. You will never know if that data is in fact unique. The reason this scenario occurred in iTAPS is that we had difficulty linking the unique identifier of the Element Title with the ASP.NET data passed from page to page through the drilldown hyperlink. To mitigate this problem we imposed business rules in our code that will not allow new Element Titles unless they are unique. With more research in this area, we are confident that this flaw could be removed.

E. CREATE SCALABLE PAGES

As discussed above in the limitations, the iTAPS pages are not entirely scalable. The usability of iTAPS could be enhanced by ensuring that the text and charts of each page can scale to different sizes for more dynamic presentations.

F. CUSTOMIZE FOR DIFFERENT BROWSERS

Personal Data Assistants (PDAs) such as iPaqs running Windows CE are becoming more and more prevalent in the workplace. In addition, wireless networking is bringing these devices online. Customized screens of iTAPS could be designed to facilitate the use of such devices to enable graders to enter grades directly from their PDAs. This might require incorporation of the Wireless Application Protocol (WAP) style of HTML.

G. AUTOMATE ENTIRE INSTALL

The installation of iTAPS is a multi-stepped process (see Appendix A, User Manual). The main steps of an installation include loading the iTAPS database, installing ChartFX, and then installing the actual iTAPS code on the web server. It would be valuable to conduct additional research to find a utility that could streamline the installation process and make it a one step installation.

H. IMPORT DOMAIN USERNAMES FOR GRADERS

iTAPS uses Windows Integrated Security. To make this work the iTAPS grader usernames must match those on the domain server. This is not a difficult requirement, and it is desirable so that a user doesn't have to remember multiple usernames and passwords. However when the grader information is entered in to iTAPS, it must be typed in manually. Further research could develop a way for this step to be automated. The result would be a feature whereby the user name was selected from the list of available user on the Domain instead of being manually typed in.

I. ADAPT TO WORK WITH CODS

Second Fleet is taking the lead in coalition networking by testing out an experimental system with multiple levels of security called the Coalition Data Server (CODS). CODS is running on a Sun Trusted Solaris Server. It appears that iTAPS could be integrated to function with this system but there are many issues that need to be examined to facilitate iTAPS integration while maintaining the functionality and security integrity of CODS.

J. ADD PICTURE AND VIDEO LINK CAPABILITY FOR GRADE INPUT

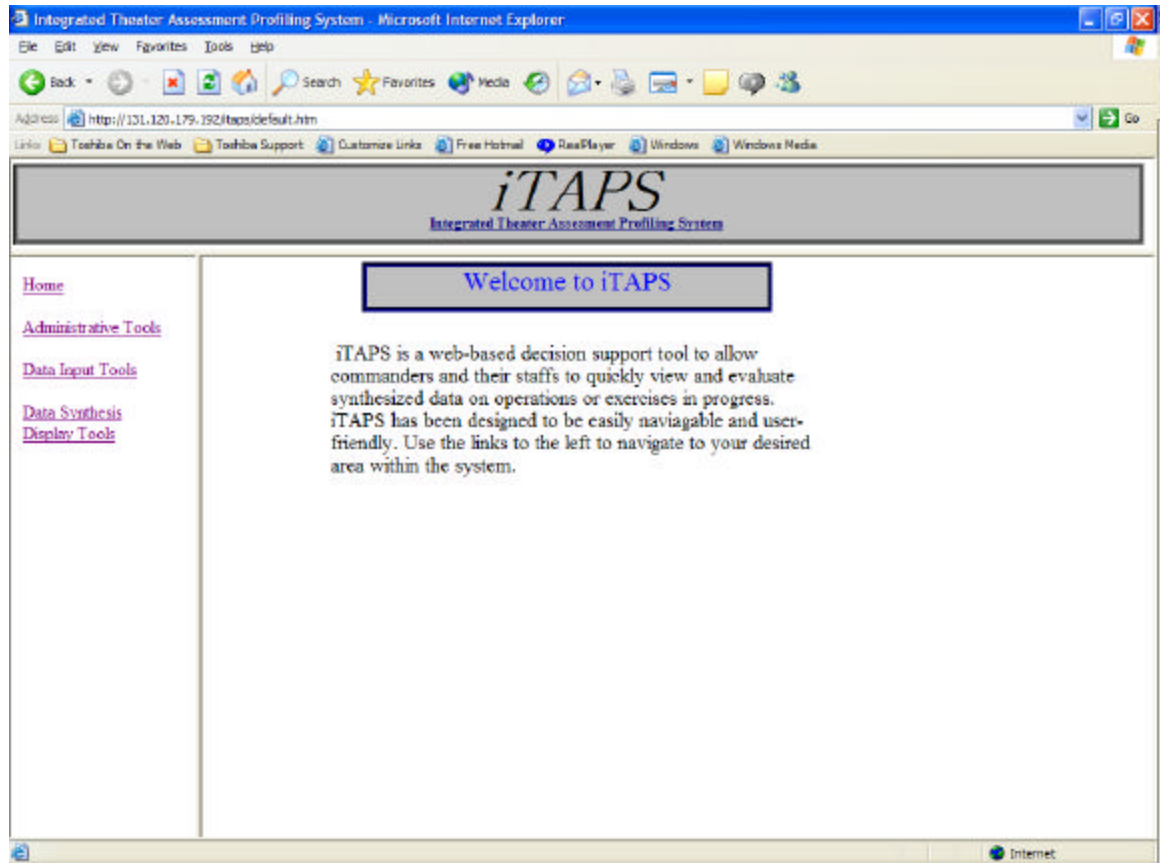
The USS Mount Whitney runs a 100 megabit fiber ATM to the desktop LAN. The bandwidth of that network, coupled with the versatile capabilities of SQL Server could easily facilitate the integration of graphics and video with iTAPS. The iTAPS database could easily be modified to include picture and video files that are related to the element grades. The end result would be that when a user drills down to a bottom level grade there would be a wealth of information provided to support the grade selection. Not only could the user now see a picture of the grader, he could also watch a video made by that person providing details of events contributing to the grade. In addition, the grader could

also supply pictures of the real world events to help explain his assessment. This could significantly enhance the capabilities of iTAPS by literally putting a face on the data it provides, as well as enhancing the quality of the data itself.

APPENDIX A ITAPS USERS MANUAL

ITAPS User's Manual

Revision 1.0



The welcome screen

1.0 Introduction

1.1. ITAPS is a powerful decision support tool for senior decision makers. It facilitates rapid aggregation of data from various levels within the organization and displays it in a manner which is readily understandable and easily searchable. To use ITAPS it is important to understand the different types of users that the ITAPS system is designed for and how the interfaces for those users work. The concept of operation of the system is also important to understand as it describes the paradigm of system operation.

2.0 Concept of Operation

2.1. ITAPS has three functional areas: an administrative area, an area for data input, and an area for data display.

2.1.1. The majority of users will interface with the Data Synthesis Display Tools. They are referred to as “view users”. This is where the display diagrams known as radar diagrams display actual and aggregated grade information on areas deemed important to monitor by the organization. These diagrams provide a drilldown capability so that more detail can be explored as the viewer desires.

2.1.2. Another class of users is the data input users. These users are authorized to access the Data Input Tools and enter grades for elements they are authorized to grade.

2.1.3. The last type of user is the administrative user. This type of user has access to the Administrative Tools and can setup the structure of the data behind the display diagrams, and can enter and authorize graders.

3.0 System Description

3.1 Functional Concepts. ITAPS is structured into Operations, Phases, Elements and Grades as its core components. Other items support these core concepts.

3.1.1 Operations. Operations are entries for the overall operation that is being graded and viewed. Operations can be active or inactive and may be linked to Phases and Elements.

3.1.1.1 Business Rules for Operations

3.1.1.1.1 Operations must have a title

3.1.1.1.2 Operations must be active for their Elements to be graded by Graders

3.1.2 Phases. Each Operation can consist of several Phases, one of which is designated the active Phase.

3.1.2.1 Business Rules for Phases

3.1.2.1.1 A Phase must have a Title

3.1.2.1.2 Only one Phase of an Operation may be active at a time

3.1.2.1.3 A Phase must be active for it's Elements to be graded by Graders

3.1.3 Elements. Each Phase can consist of an unlimited number of Elements in whatever hierarchy the user desires. These Elements may be active or inactive. Several business rules govern Elements as they are the primary core component of ITAPS.

3.1.3.1 Business Rules for Elements

3.1.3.1.1 An Element must have an Element Title

3.1.3.1.2 An Element must be linked to an Operation, Phase and Parent Element to exist

3.1.3.1.3 The "Master Element" of an Operation/Phase represents the Phase

3.1.3.1.4 Two Elements in the same Phase with the same Sidecolor cannot have the same Element Title

3.1.3.1.5 Elements must be active to be visible to view users or graded by Graders

- 3.1.3.1.6 Once an Element is created, it's Sidecolor cannot be changed
 - 3.1.3.1.7 If an Element has children, it must have at least three children for the Radar Diagrams to display correctly. If your Element only has one Child Element, replace that Element with the Child. If your Element has two Children, create a "dummy" active Child Element whose grade weight is zero
 - 3.1.3.1.8 Elements with children should be designated as "auto-graded"
 - 3.1.3.1.9 Elements without child Elements are considered "Bottom" Elements and must have a designated grader or the Radar Diagrams will not calculate grades correctly
- 3.1.4 Element Titles. Element Titles are entered separately from Elements because there is significant reuse of Element Titles, therefore separating the two minimizes reentry of identical data and facilitates data mining for post-action, cross operation analysis.
- 3.1.5 Point-of-Contact(POC). POC entries are for giving the data input users a point-of-contact to ask questions about and receive help on ITAPS use for their functional area.
- 3.1.6 Grader Category. Grader Categories are for segregating Graders into functional areas and linking POC's to Graders to for that functional area. Example: JFACC would be a Grader Category.
- 3.1.7 Graders. Graders are the personnel who will be the Data Input Users.
- 3.1.8 Grades. Grades are entered by Graders for Elements that they are authorized to grade. Grades are auto generated by ITAPS for Elements without direct Graders. This Gradedata exists until the Operation/Phase/Element they are linked with are deleted.
- 3.1.9 Side Color. Side Color is the designated color for the participants in an Operation. It is left to the user to designate what the colors represent.

4.0 Software Installation

4.1 System Requirements

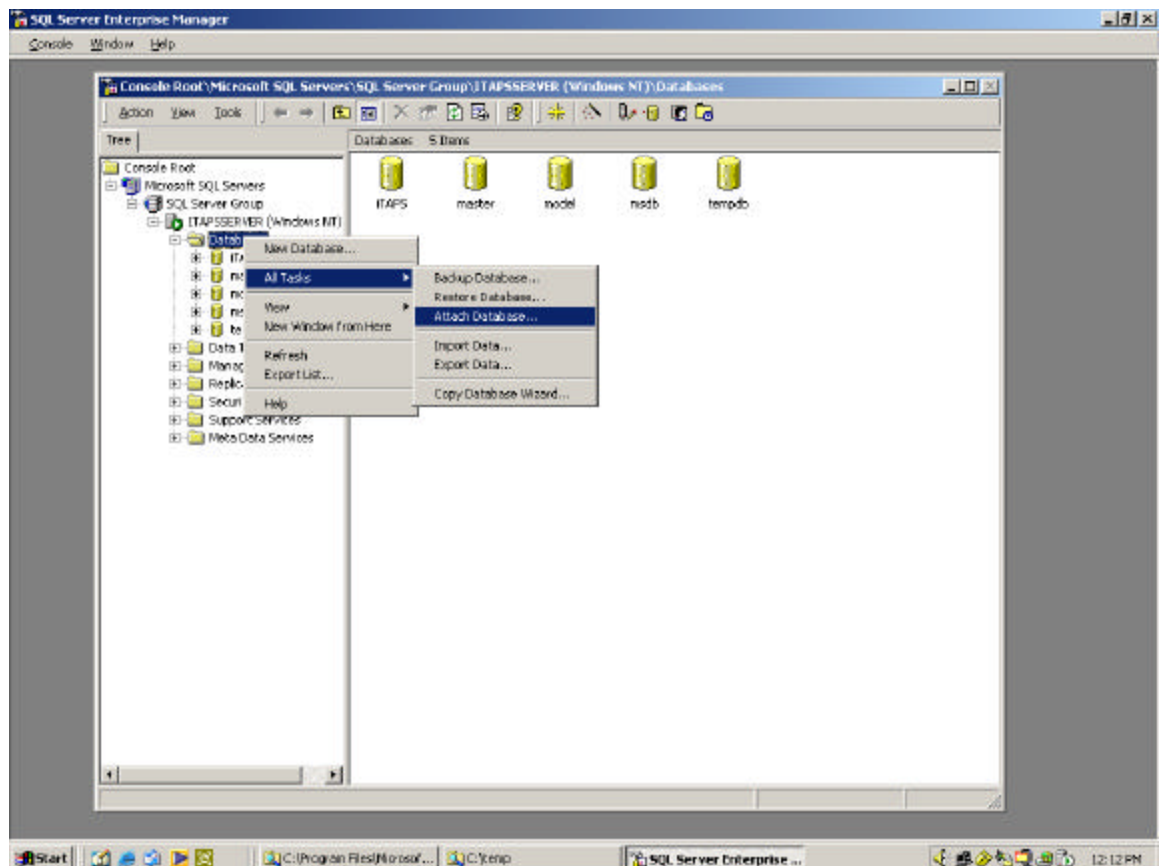
- 4.1.1 CPU: P4 1.5Ghz or greater
- 4.1.2 HD: 40GB or greater
- 4.1.3 OS: Windows 2000 Server Service Pack 3 Build 5.00.2195 or greater
- 4.1.4 DB: SQL Server 2000 version 8.00.534
- 4.1.5 NET Framework: installed and updated thru Version 1.0.3705
- 4.1.6 Web Server: Internet Information Server(IIS) version 5 running in C:\inetpub. Ensure that all IIS directories and subdirectories have anonymous access disabled and are using windows integrated permissions only.
- 4.1.7 Video requirements: This application is designed to run in 1024x768 or greater screen resolution

4.2 Application Installation

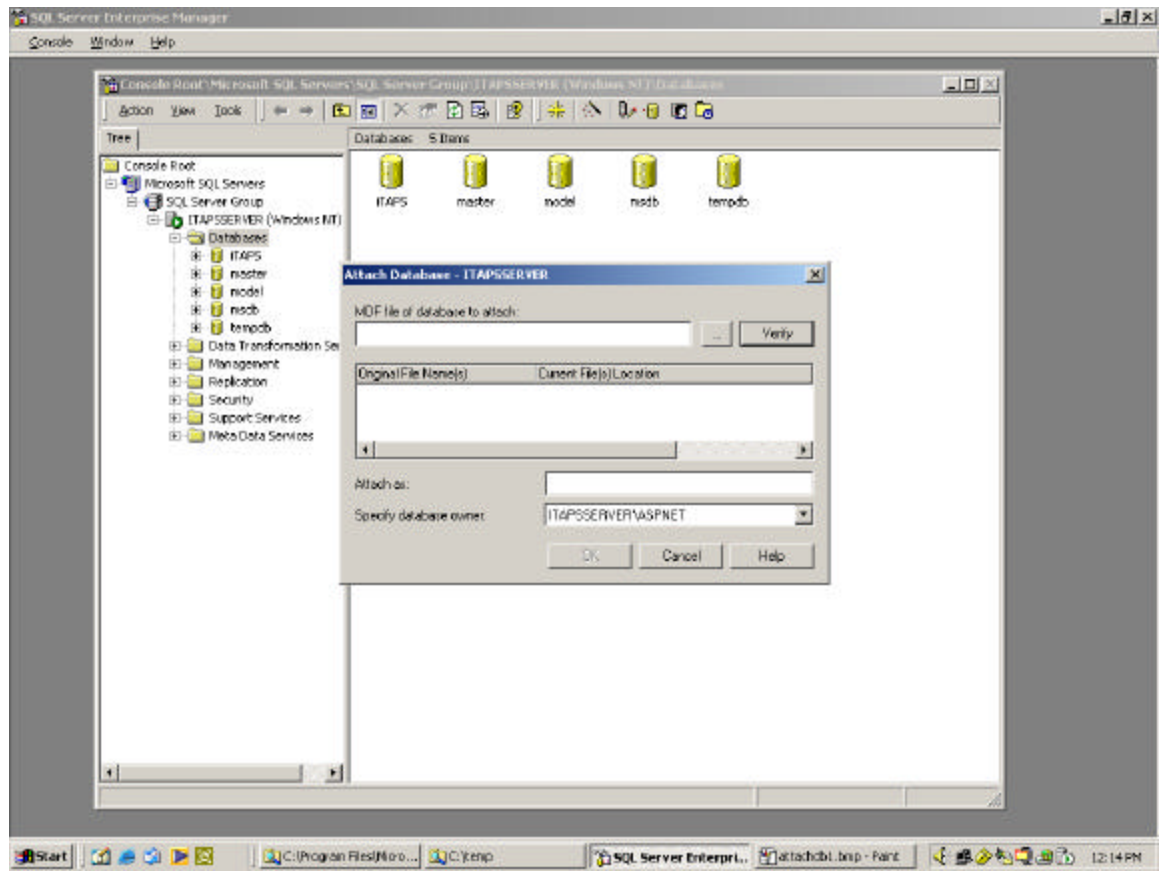
- 4.2.1 Run "Setup.exe" in the CDROM root directory (D:\ on most computers)
- 4.2.2 Follow the onscreen prompts.
- 4.2.3 Now go to the CDROM directory "ChartFX" and run "CfxNetProduction.exe". The serial number needed is in the readme of the same directory as well as the procedure to manually obtain the license key.

4.3 Database installation

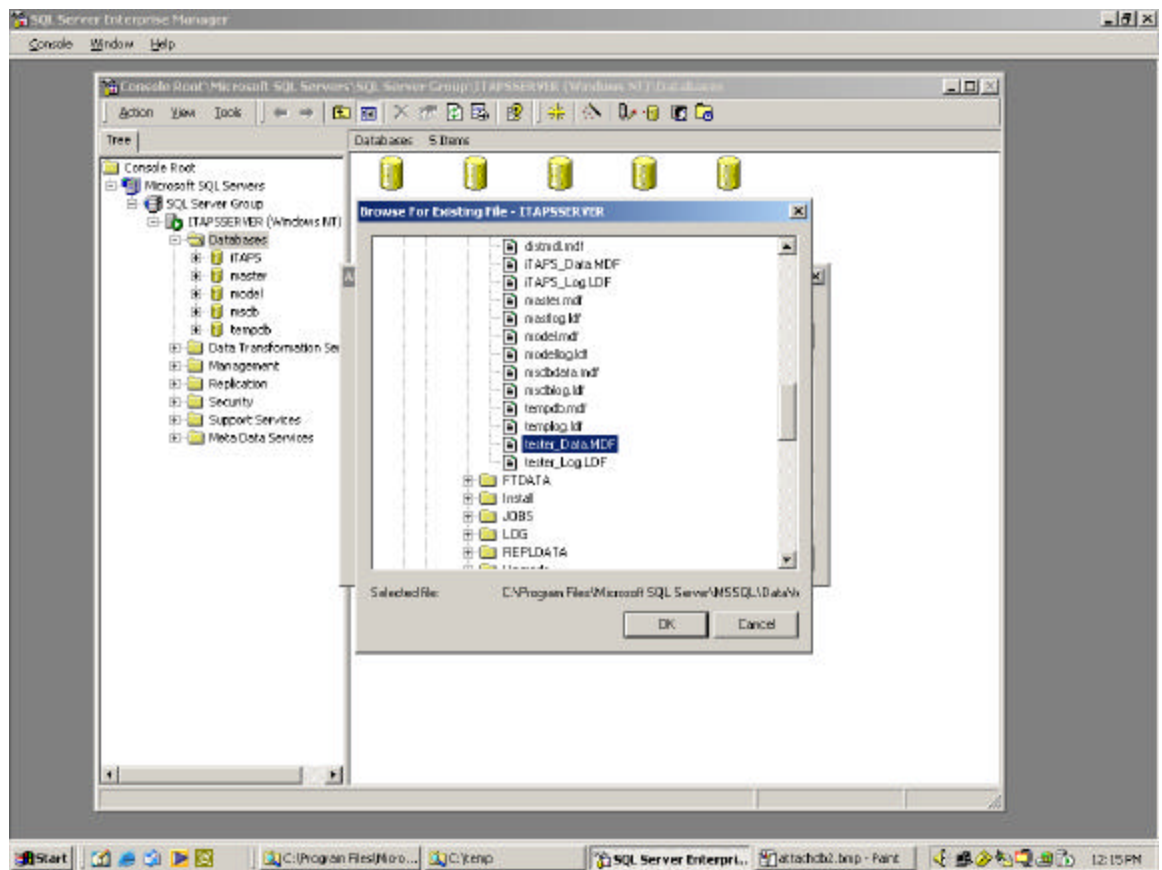
- 4.3.1 Run DBcopy.bat in the CDROM "SQL" directory (D:\SQL on most computers) to copy the database file to your computer
- 4.3.2 Use Enterprise manager to attach the ITAPS database



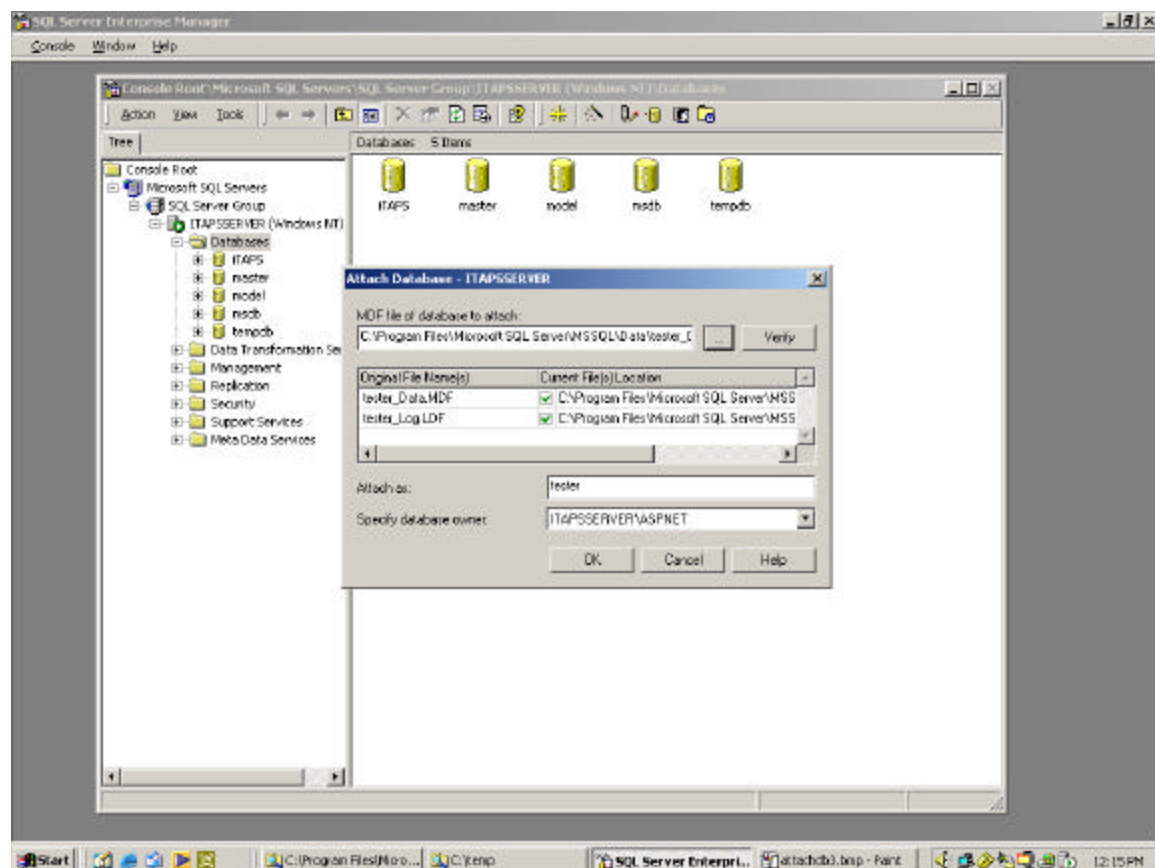
Right Click On “Databases” and select “All Tasks\Attach Database”.



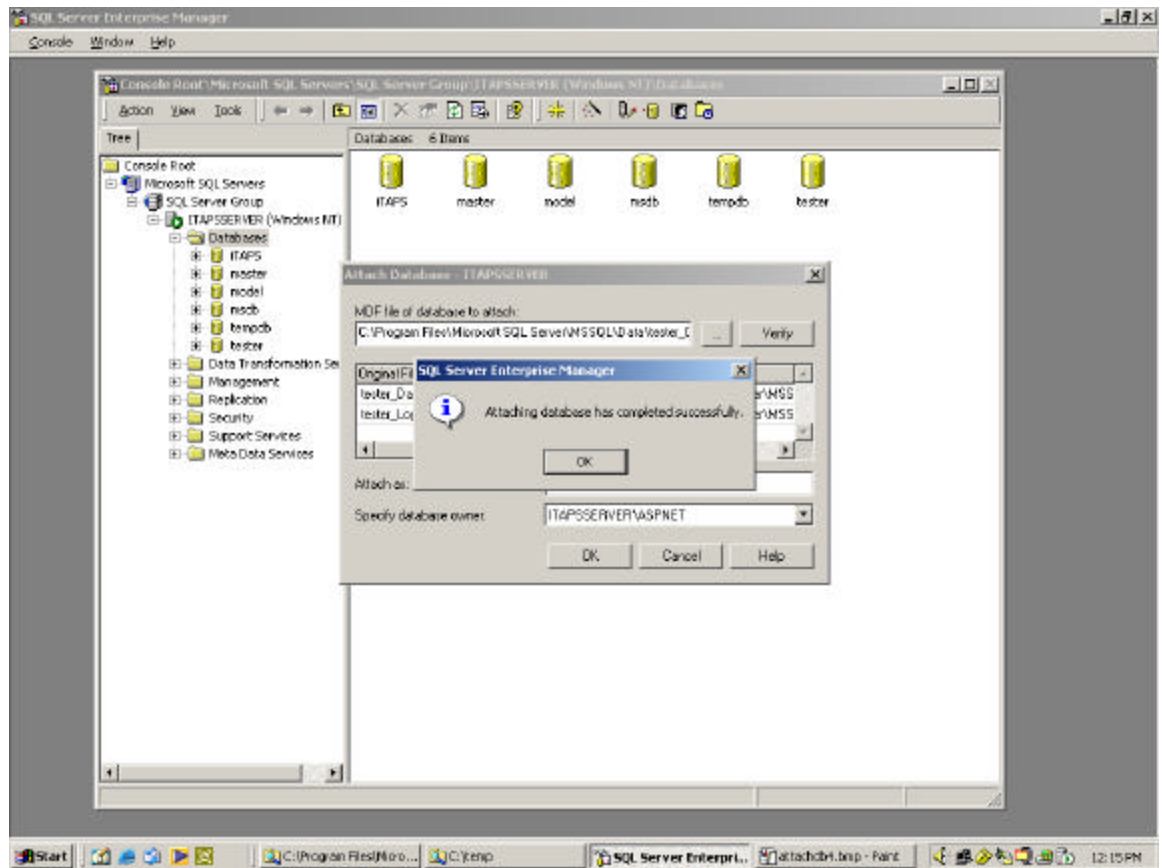
Click the “..” button adjacent to the “Verify” button to browse for the database files.



Traverse the file menu to get to “C:\Program Files\Microsoft SQL Server\MSSQL\Data\ITAPS_Data.MDF”. Click “OK”.



Click “OK” to attach the database.



If everything worked properly you should see a screen similar to the above. Click “OK” to finish attaching the database.

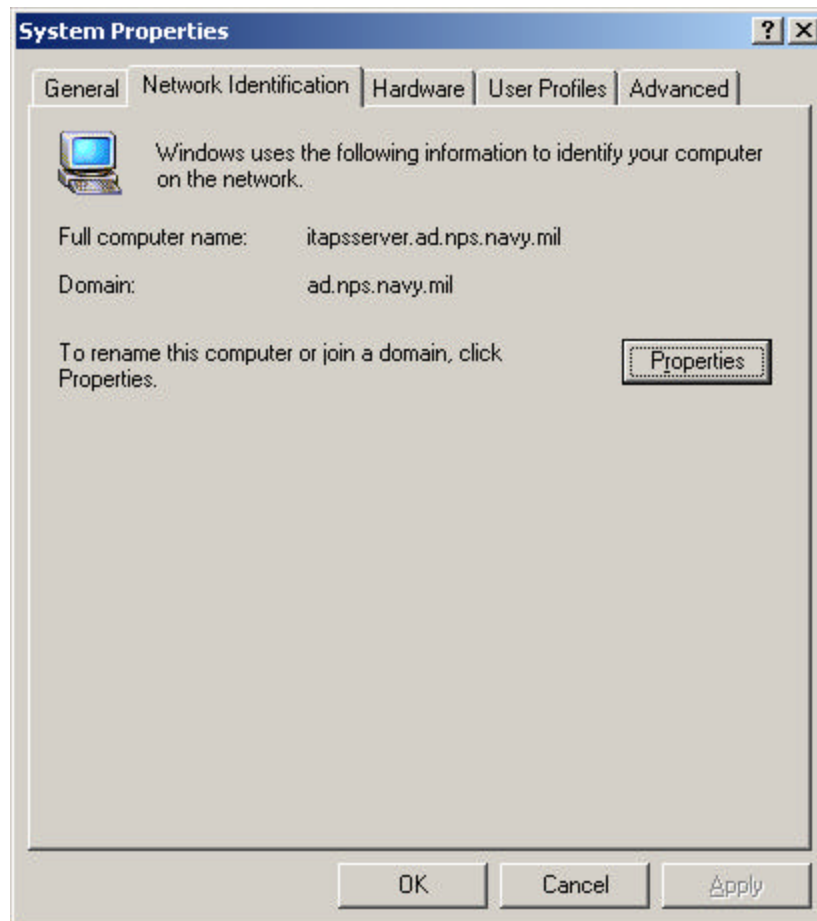
4.4 Integrated Security Setup

While all possible security settings and potential security holes are beyond the scope of this application installation, it is recommended that the server administrator run the Microsoft Baseline Security Analyzer (<http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/Tools/mbsahome.asp>) after the entire server setup process. ITAPS requires the following Security Settings to operate correctly:

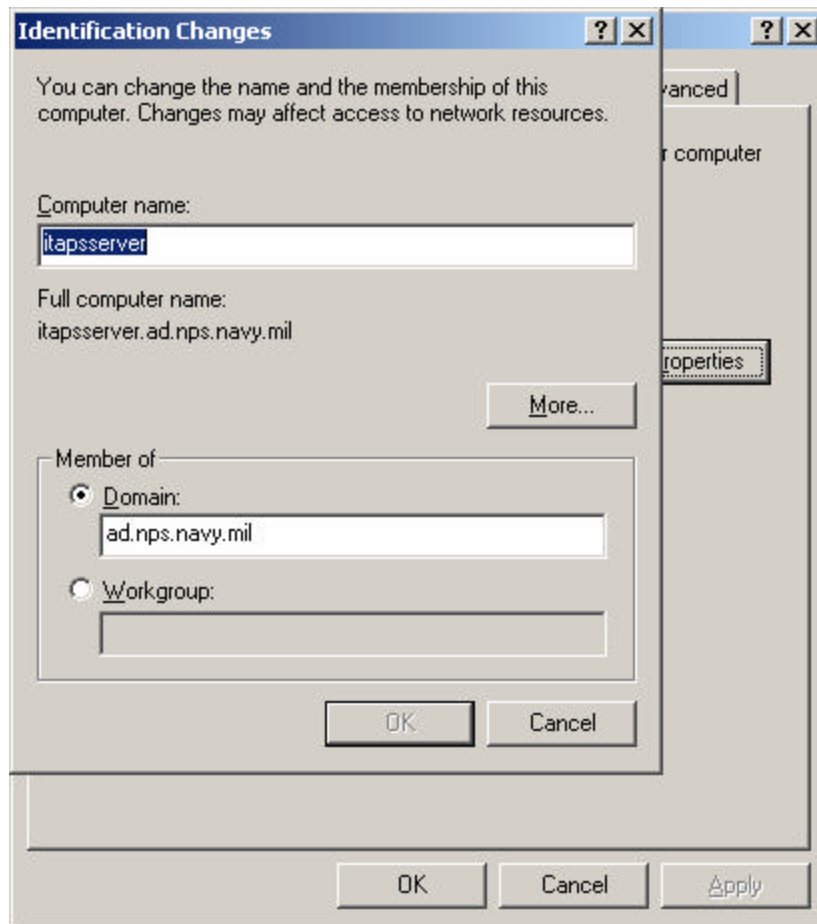
4.4.1 Windows 2000 Server

4.4.1.1 Connect this server to your domain server(s).

- Right Click on “My Computer”
- Select the “Network Identification” Tab

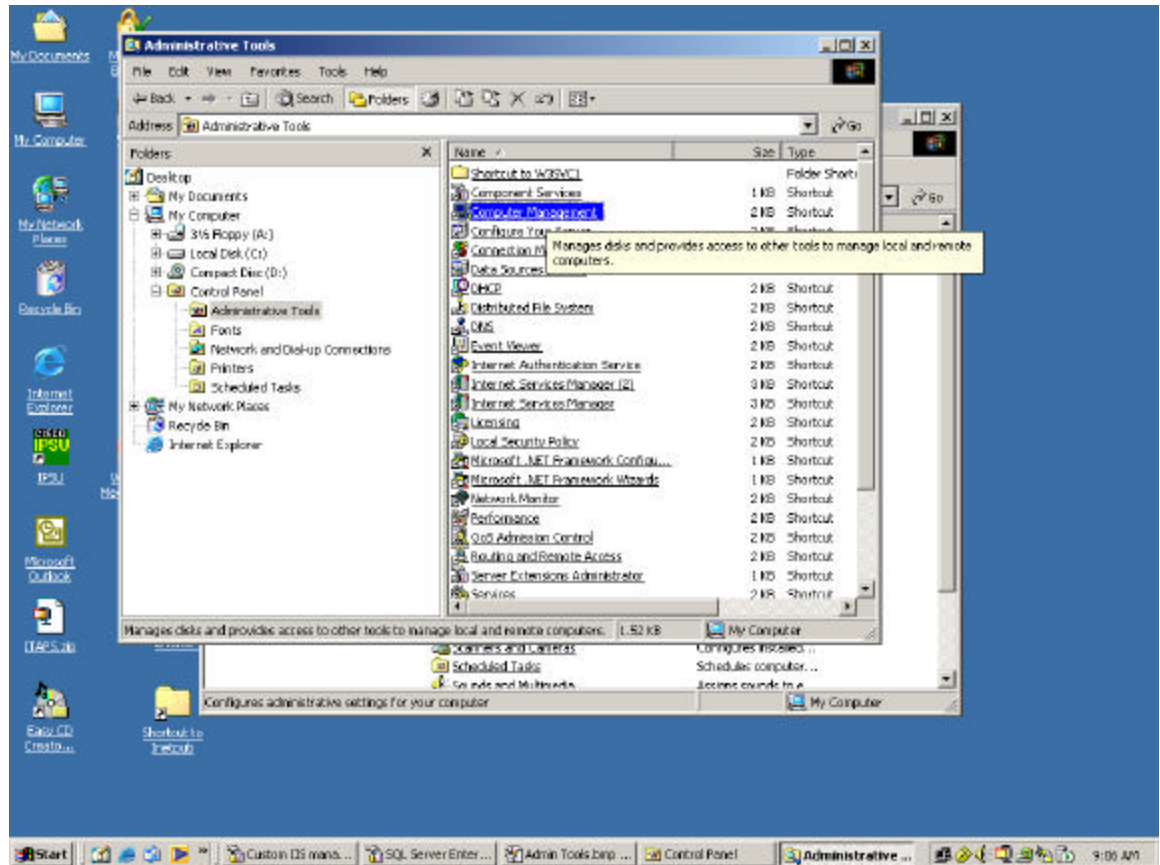


- Click “Properties”

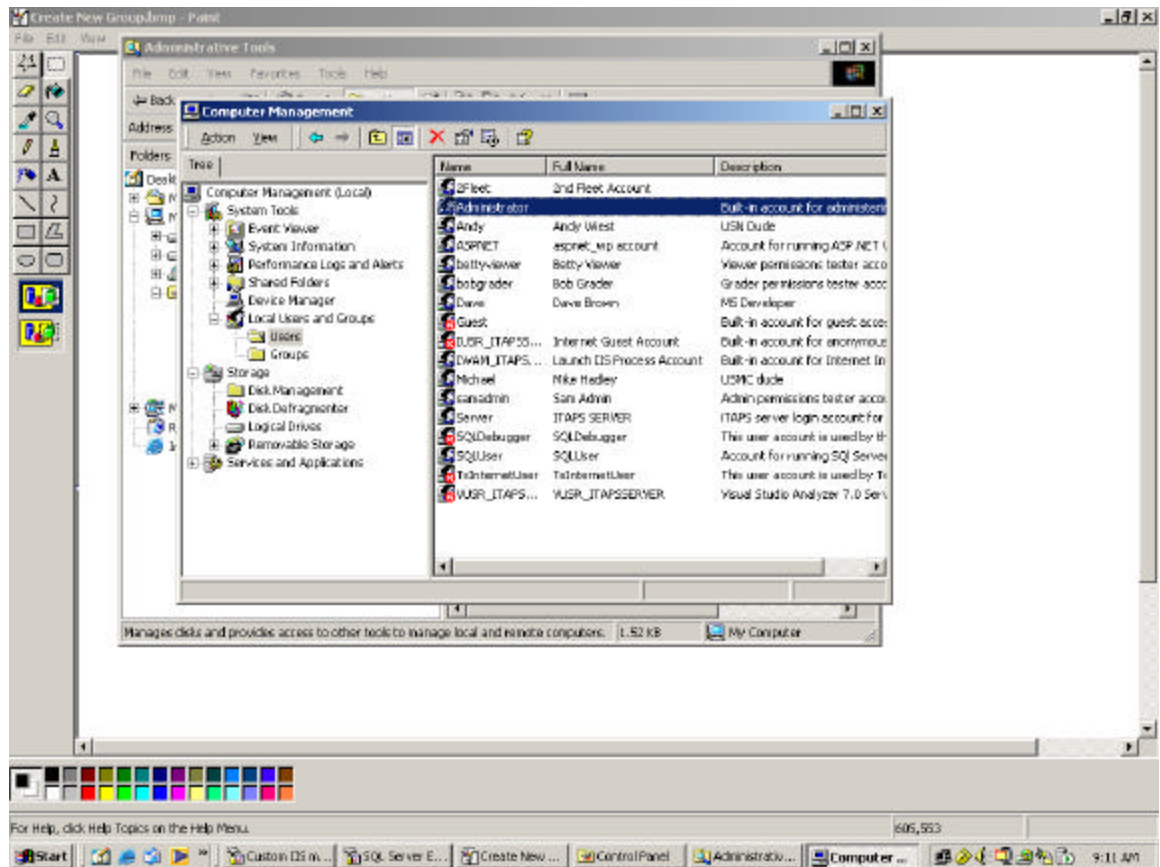


- Name this computer “ITAPSSERVER”
- Enter the name of the domain you wish to join

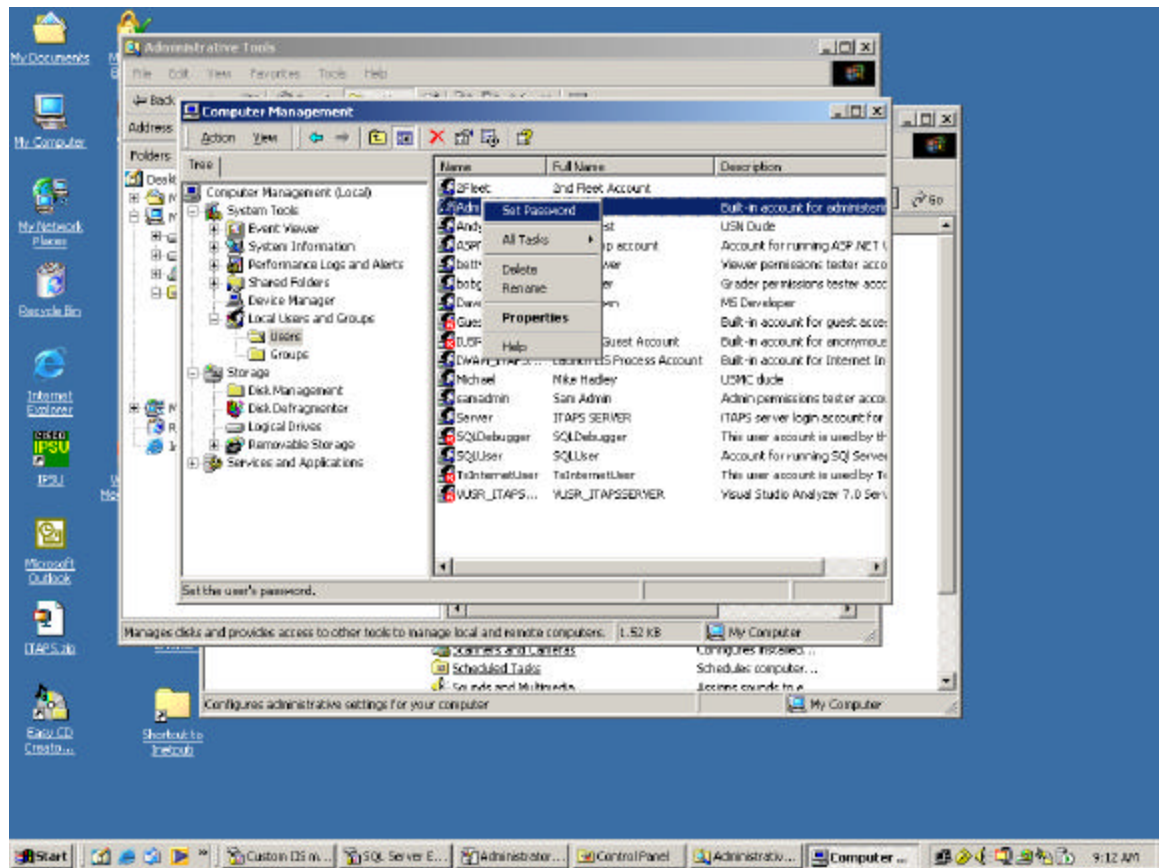
4.4.1.2 Create strong account passwords for the administrator account for ITAPSSERVER



- Open “Computer Management” (Start\Control Panel\Administrative Tools)

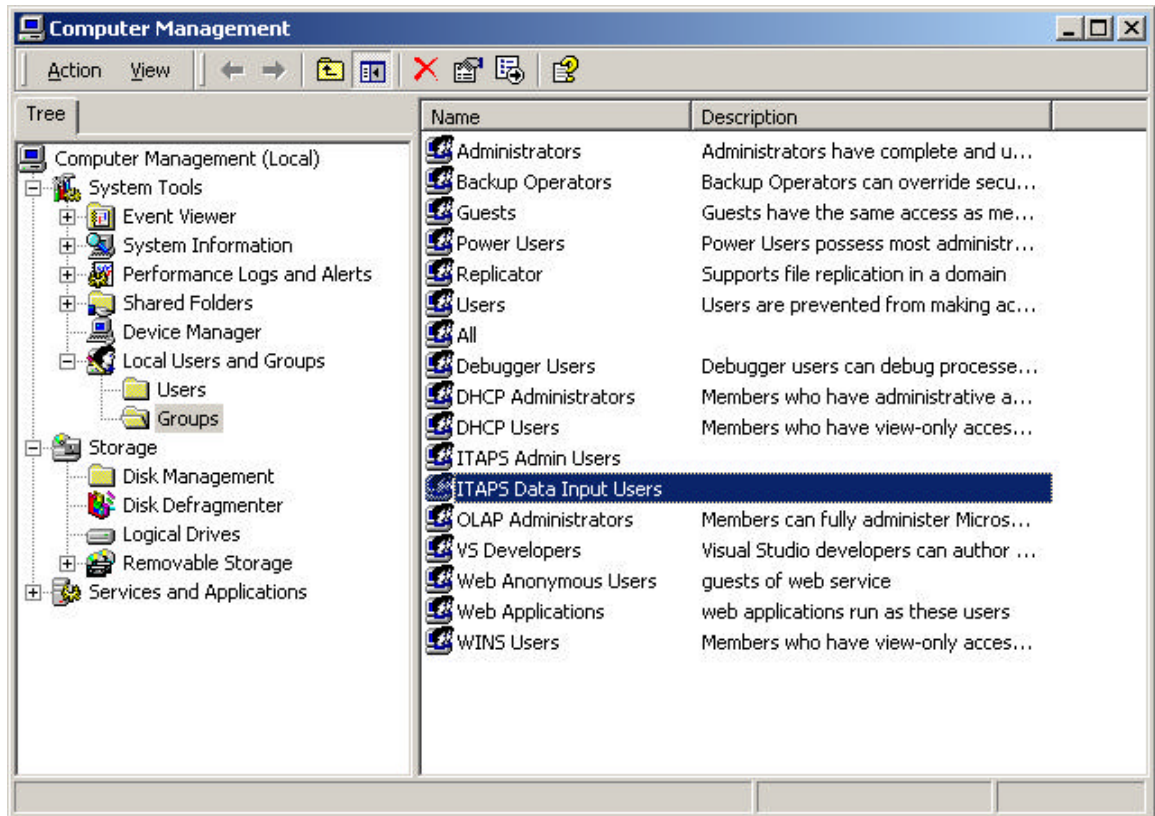


- Select “Local Users and Groups/Users” and right click the “Administrator” account.



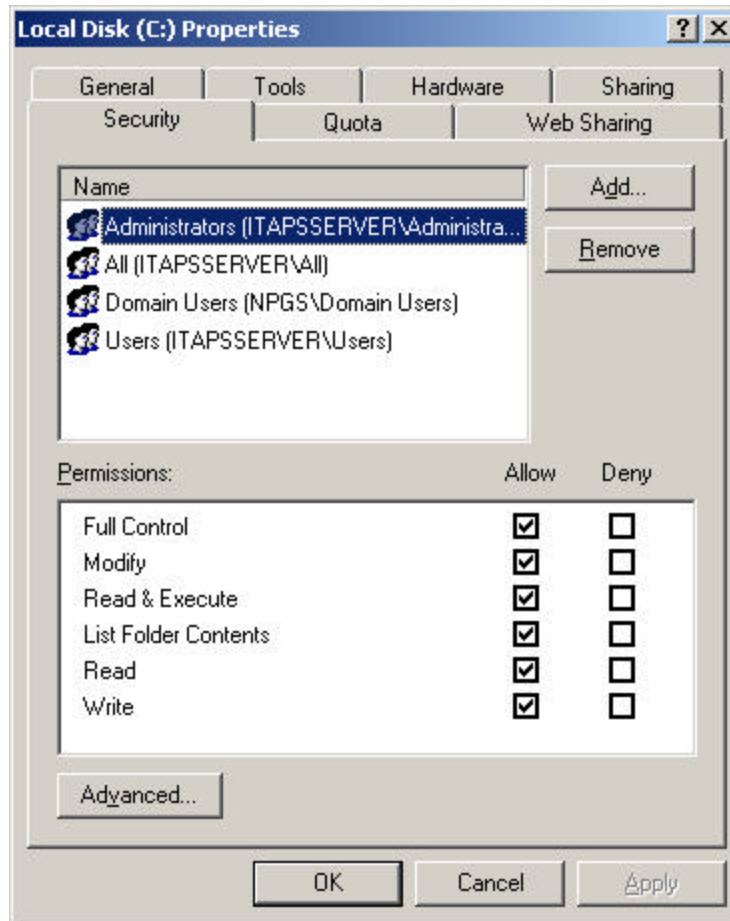
- Set the password for this account to a strong password

4.4.1.3 Create New Account Groups



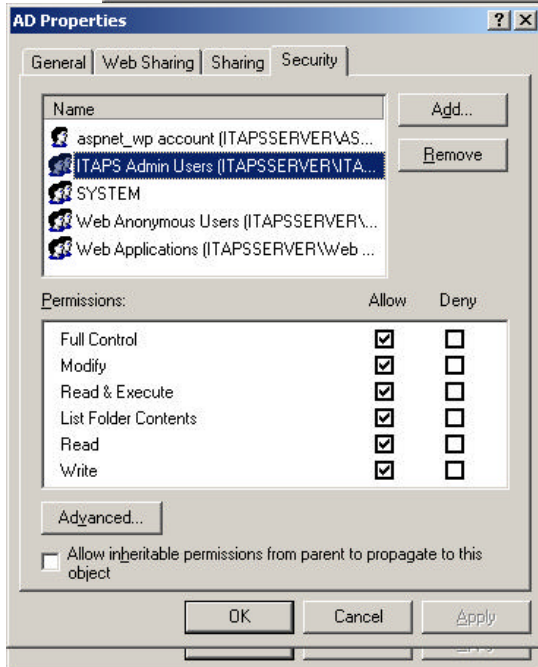
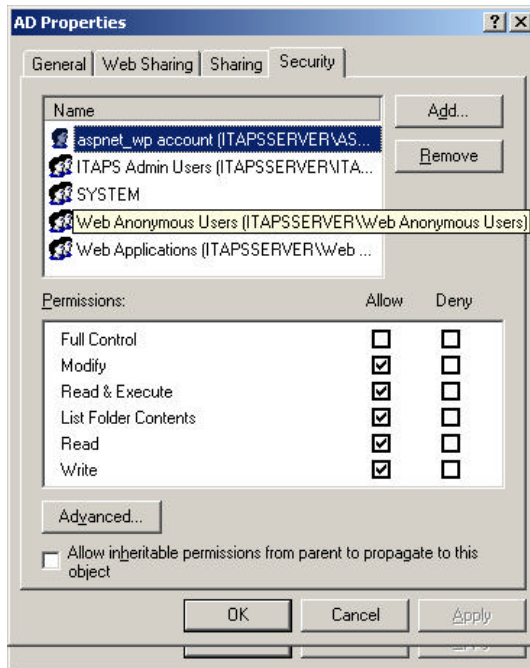
- Create three new Account Groups named “ITAPS Admin Users”, “ITAPS Data Input Users” and “All”.
- The “All” Group should only contain “NT Authority\Authenticated Users”.
- The “ITAPS Admin Users” Group should contain those logins you wish to designate as ITAPS Administrators.
- The “Data Input Users” Group should contain those logins that will function as graders

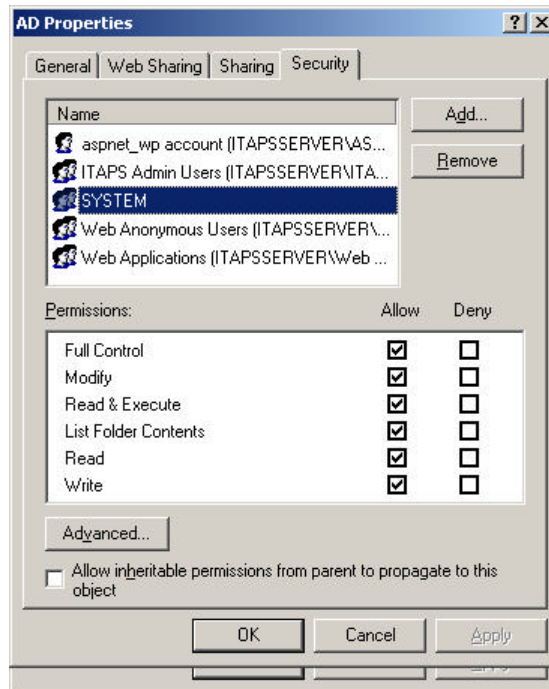
4.4.1.4 Set Permissions for the ITAPSSERVER



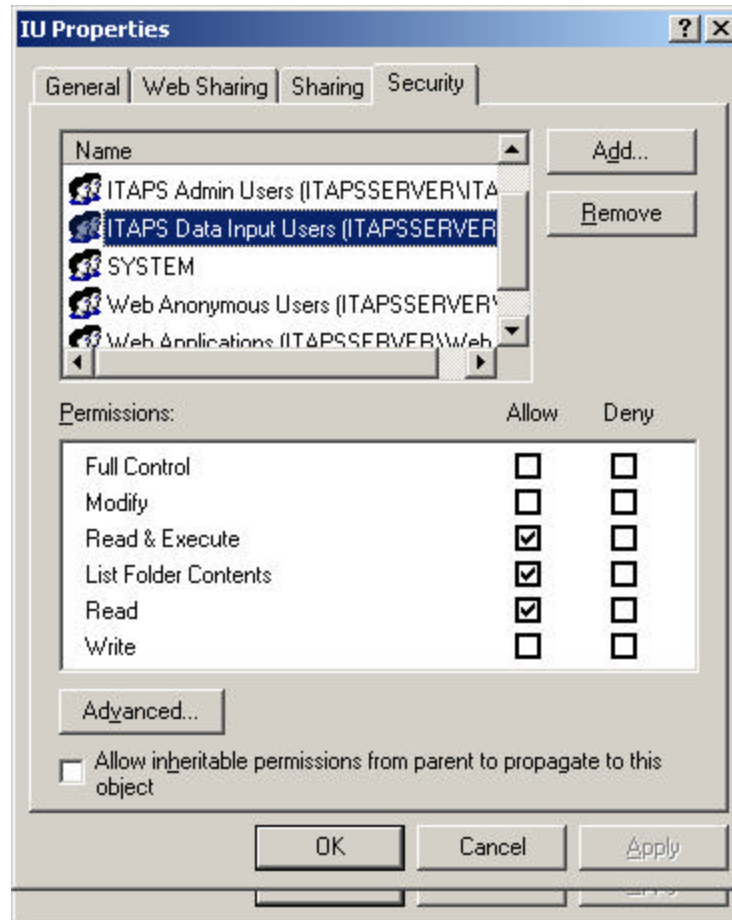
Allow "All" to access the C: drive. These permissions should propagate to every directory.

Only allow the following Groups and users with the indicated settings to access the C:\inetpub\wwwroot\ITAPS\AD directory. You will probably have to clear the “allow inheritable permissions...” check box to enable editing of these settings.

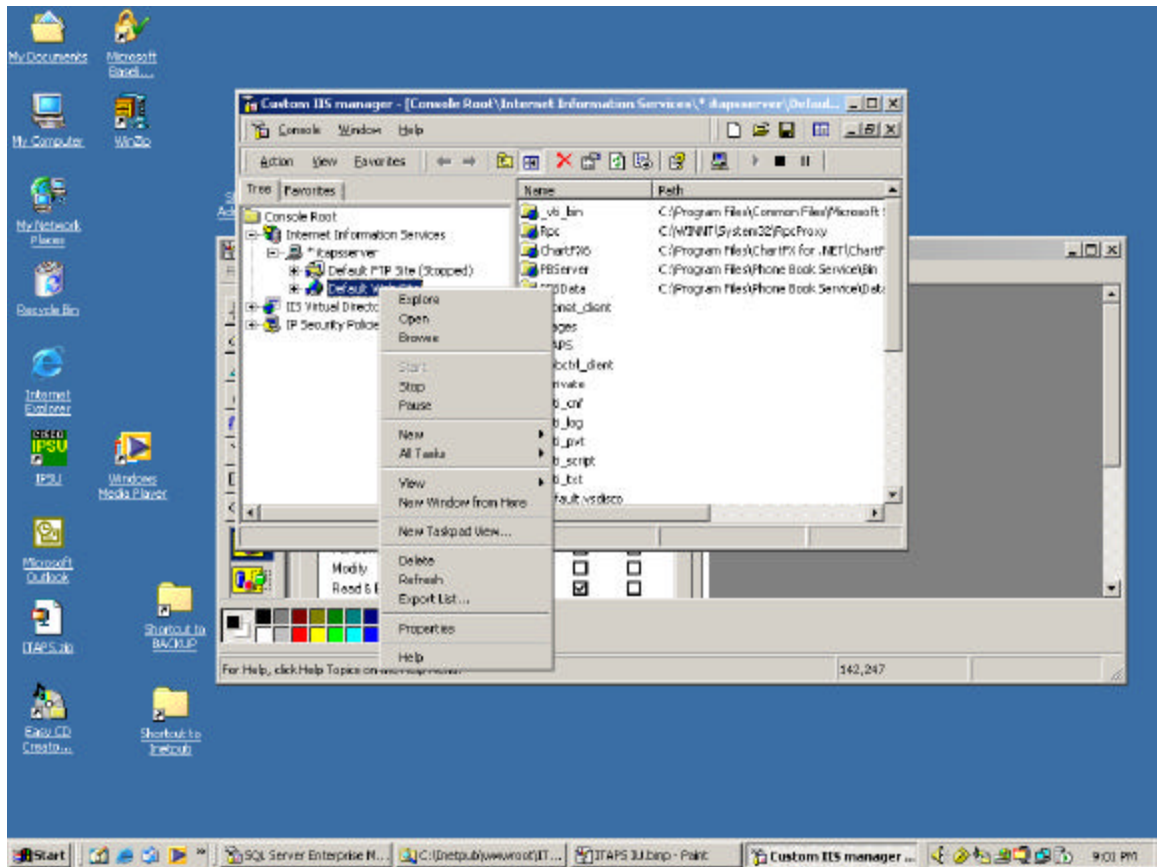




Allow the same Groups and users as above to access the C:\inetpub\wwwroot\ITAPS\AD directory. In addition add the “ITAPS Data Input Users” Group illustrated below. You will probably have to clear the “allow inheritable permissions...” check box to enable editing of these settings.



4.4.1.5 Set permissions for the ITAPSSERVER IIS directories



- Open Internet Services Manager (Start\Control Panel\Administrative Tools)
- Right Click on “Default Web Site”
- Select “Properties”

Default Web Site Properties [?] [X]

Directory Security | HTTP Headers | Custom Errors | Server Extensions
Web Site | Operators | Performance | ISAPI Filters | Home Directory | Documents

Web Site Identification

Description:

IP Address: [v]

ICP Port: SSL Port:

Connections

☒ Unlimited
☐ Limited To: connections

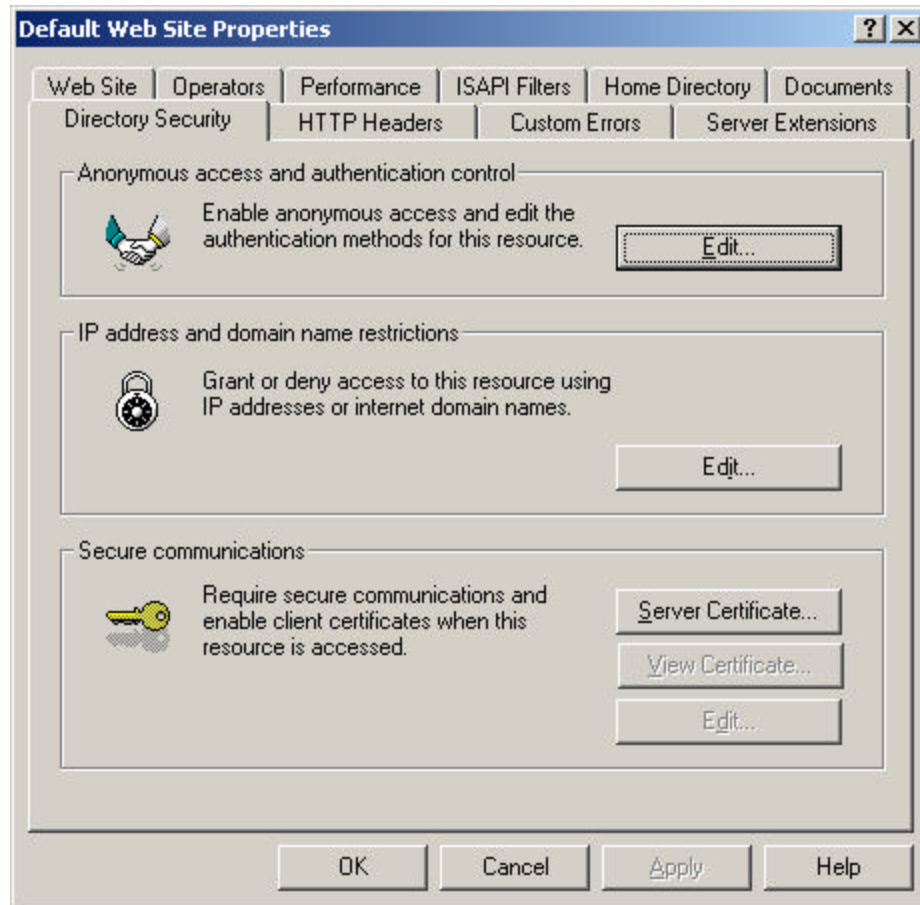
Connection Timeout: seconds

☒ HTTP Keep-Alives Enabled

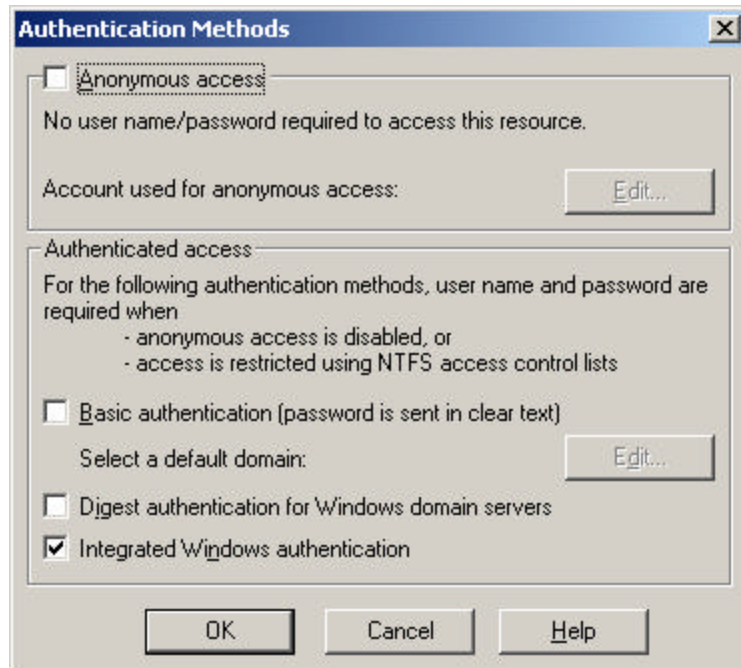
☒ Enable Logging

Active log format:
 [v]

- Select "Directory Security"



Select "Edit"



Make your settings identical to this picture then Click "OK". Click "OK" on the main settings screen, then close IIS Manager.

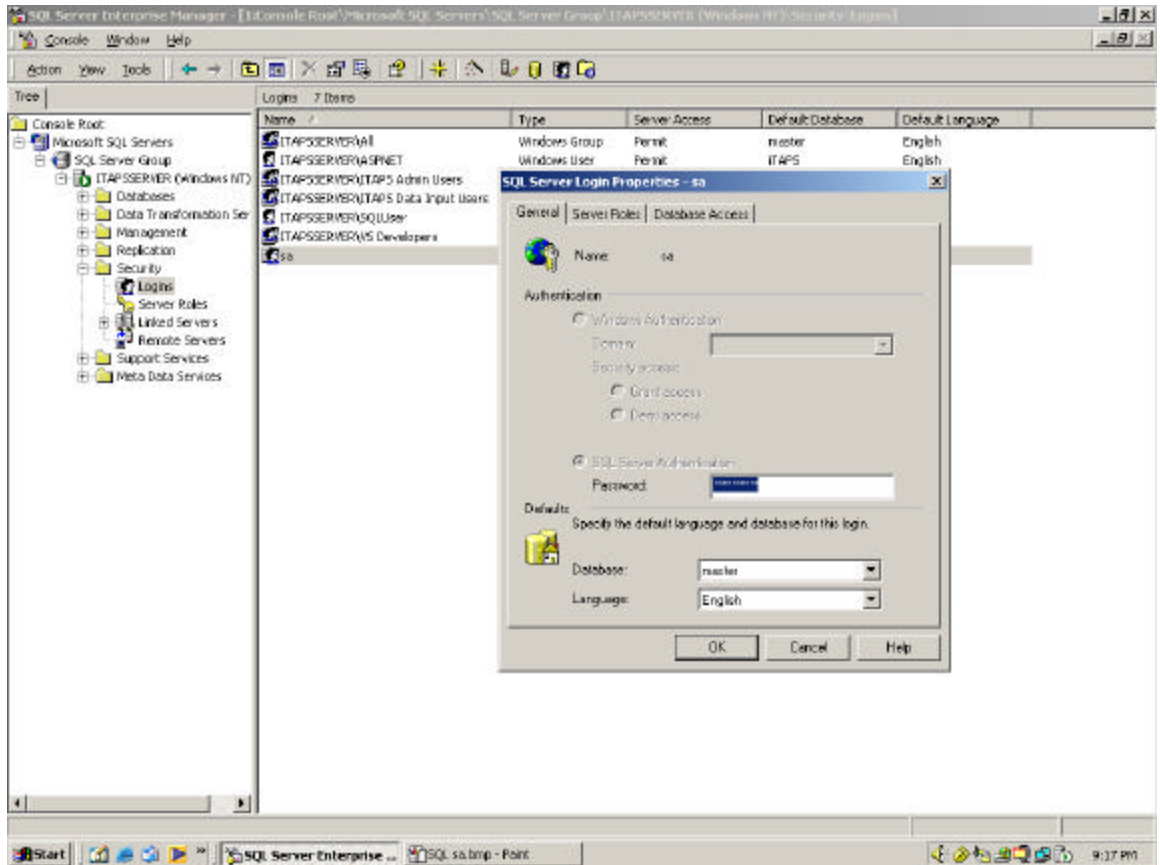
Your Windows 2000 Server permissions should now be set properly. If you have any problems, contact your system administrator.

It is further recommended that you turn off the ftp and nntp servers running under IIS. Contact your system administrator on how to accomplish this.

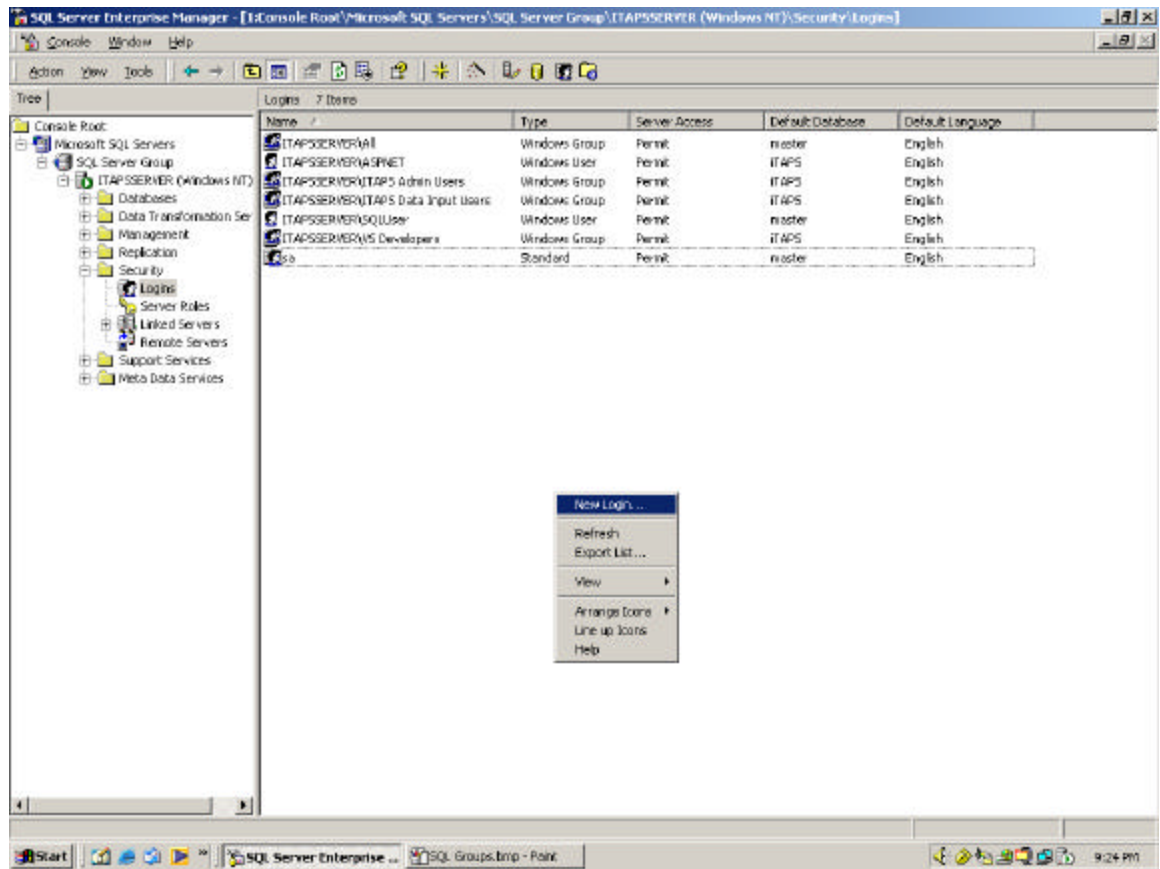
4.4.2 SQL Server

- 4.4.2.1 Enter a strong password for the sa account. In Enterprise Manager go to “Security/Logins” and create a strong password for the sa account

4.4.2.2

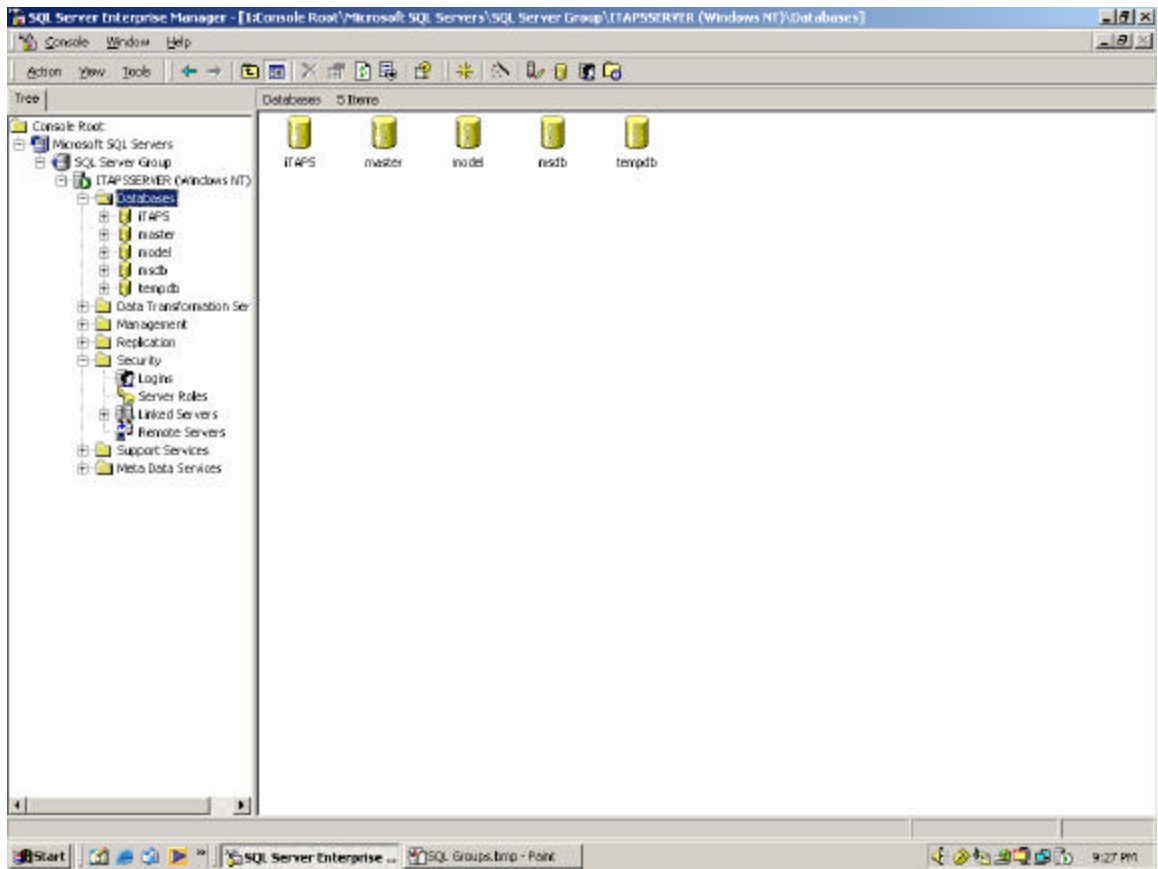


4.4.2.3 Set SQL Server 2000 global login settings



Add the following Groups to your SQL Server Logins by right clicking in the open area and selecting "New Login": All, ASPNET, ITAPS Admin Users, and ITAPS Data Input Users.

4.4.2.4 Remove all non-essential databases

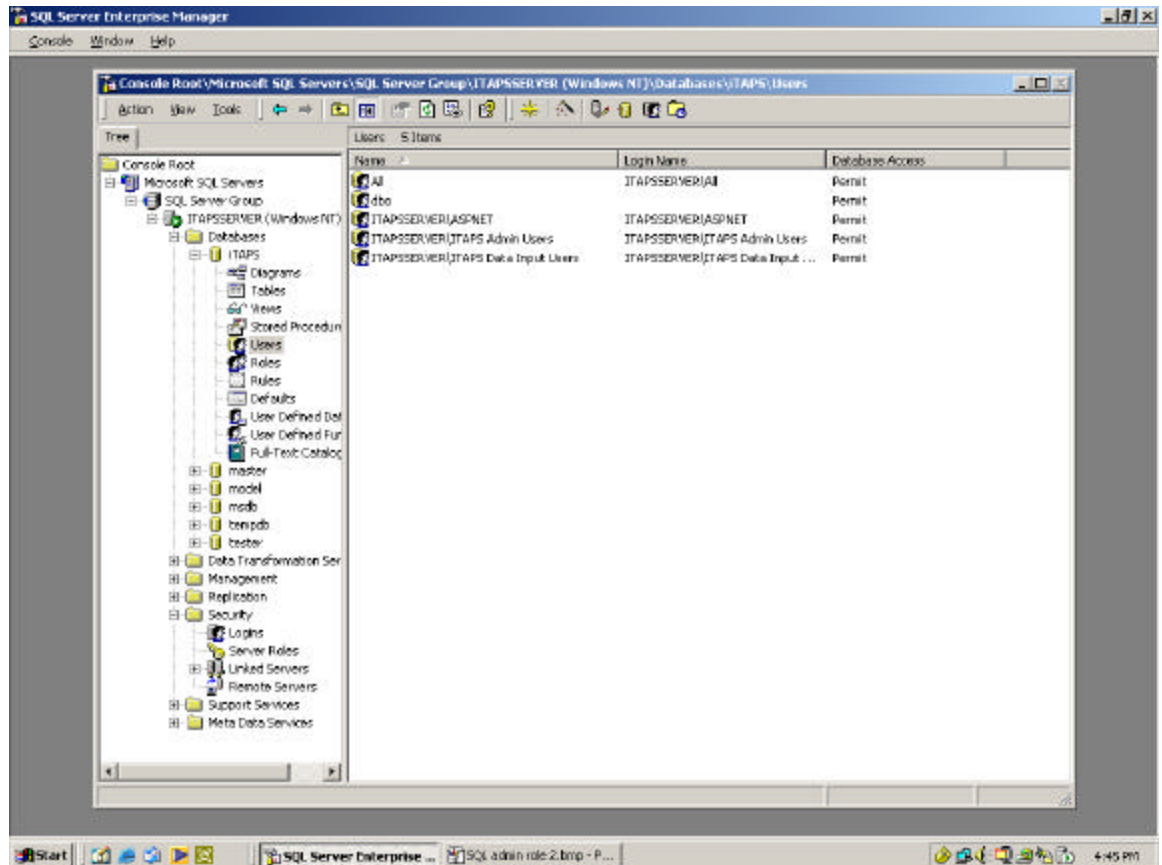


Remove all databases except the following: ITAPS, master, model, msdb and tempdb.

4.4.2.5 Create Database Roles

Database roles should have been imported with the database install.

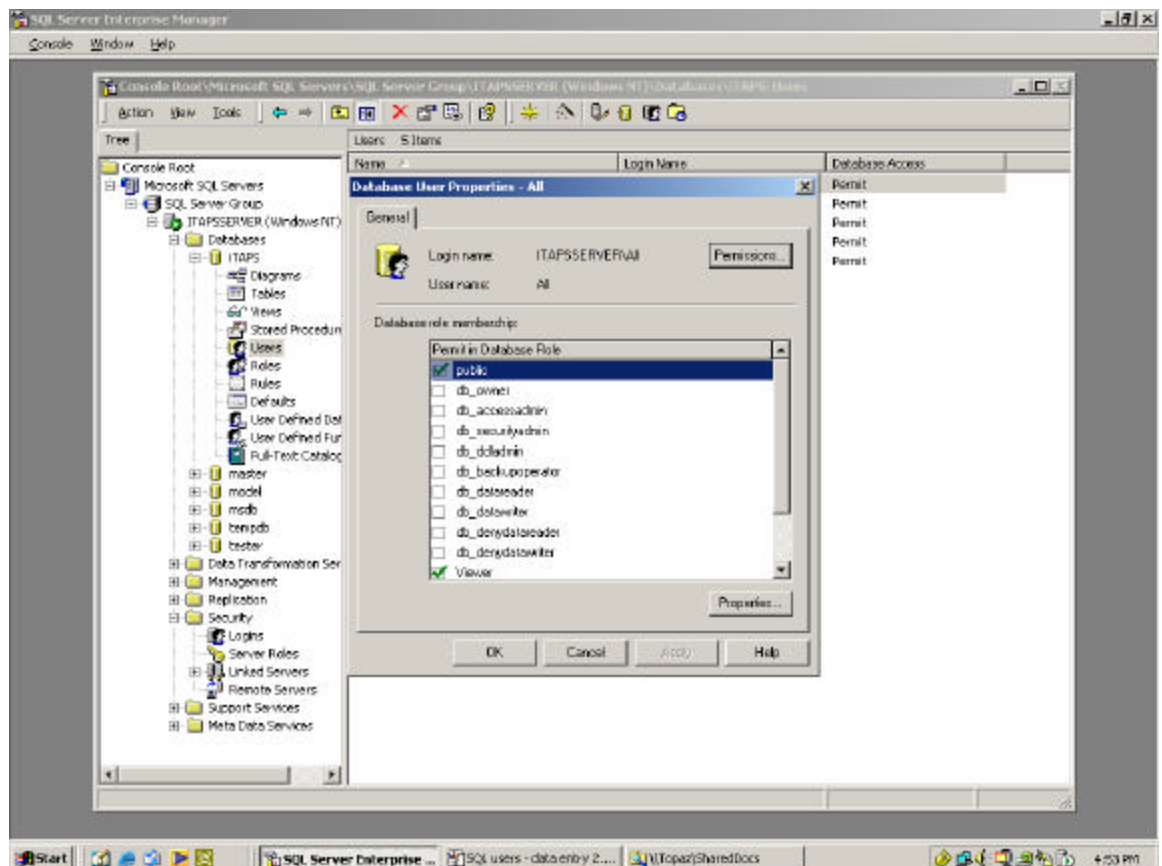
4.4.2.6 Create Database Users and link roles



Create/verify the following users from the Logins you created previously:

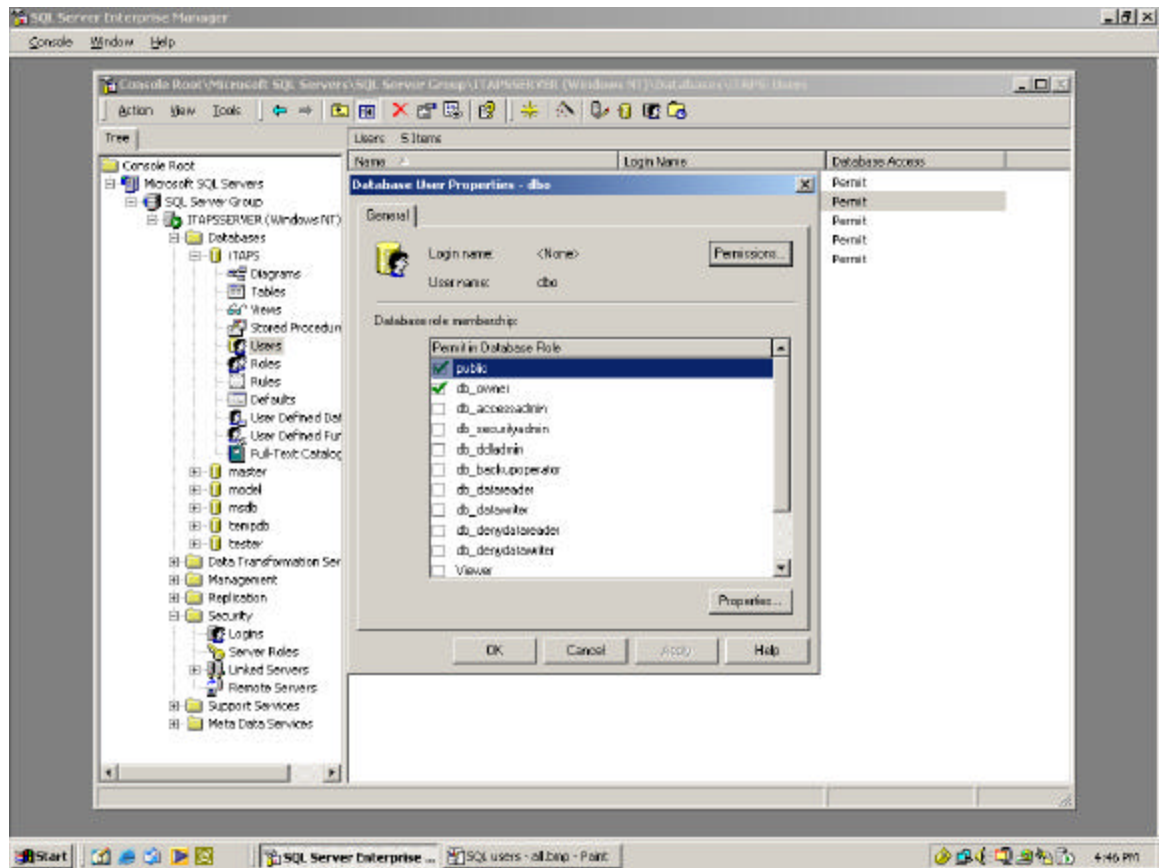
- All
- dbo
- ASPNET
- ITAPS Admin Users
- ITAPS Data Input Users

This step links the logins you created to the ITAPS database specifically.



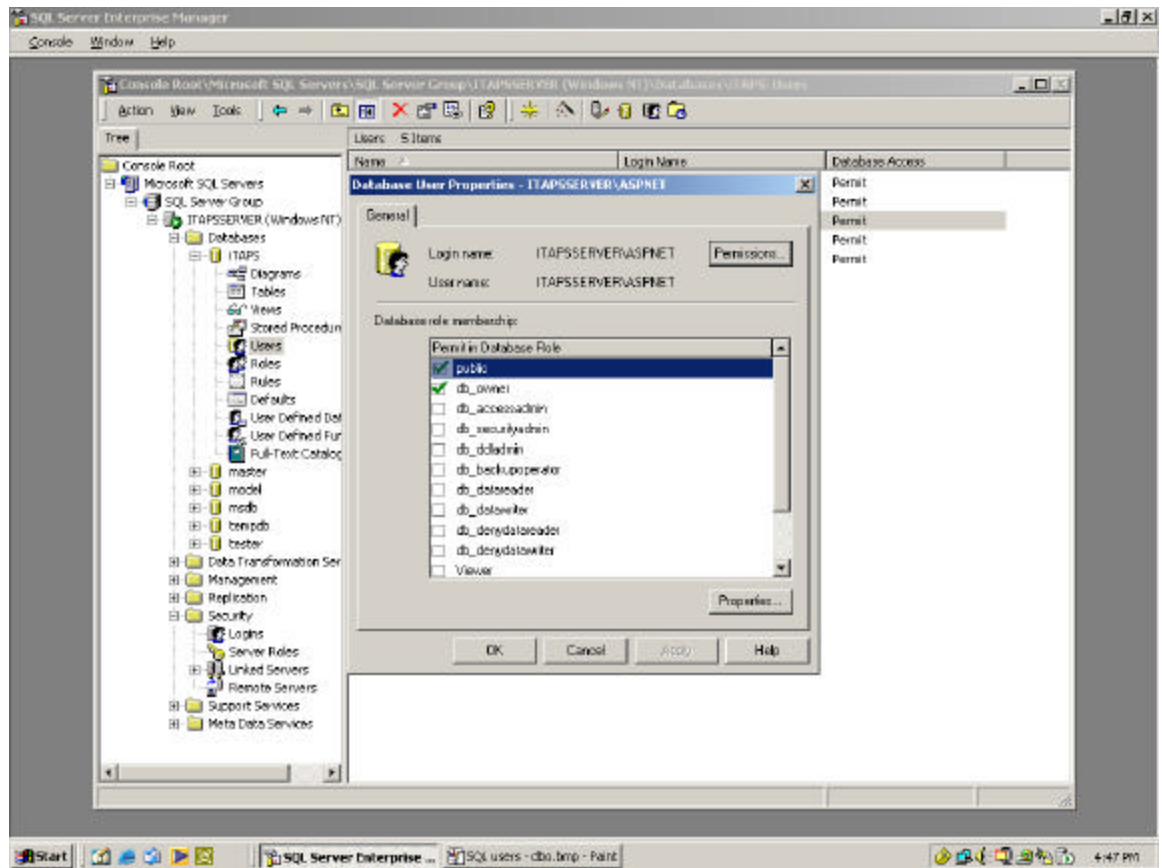
The “ALL” user should have the following roles:

- public
- Viewer



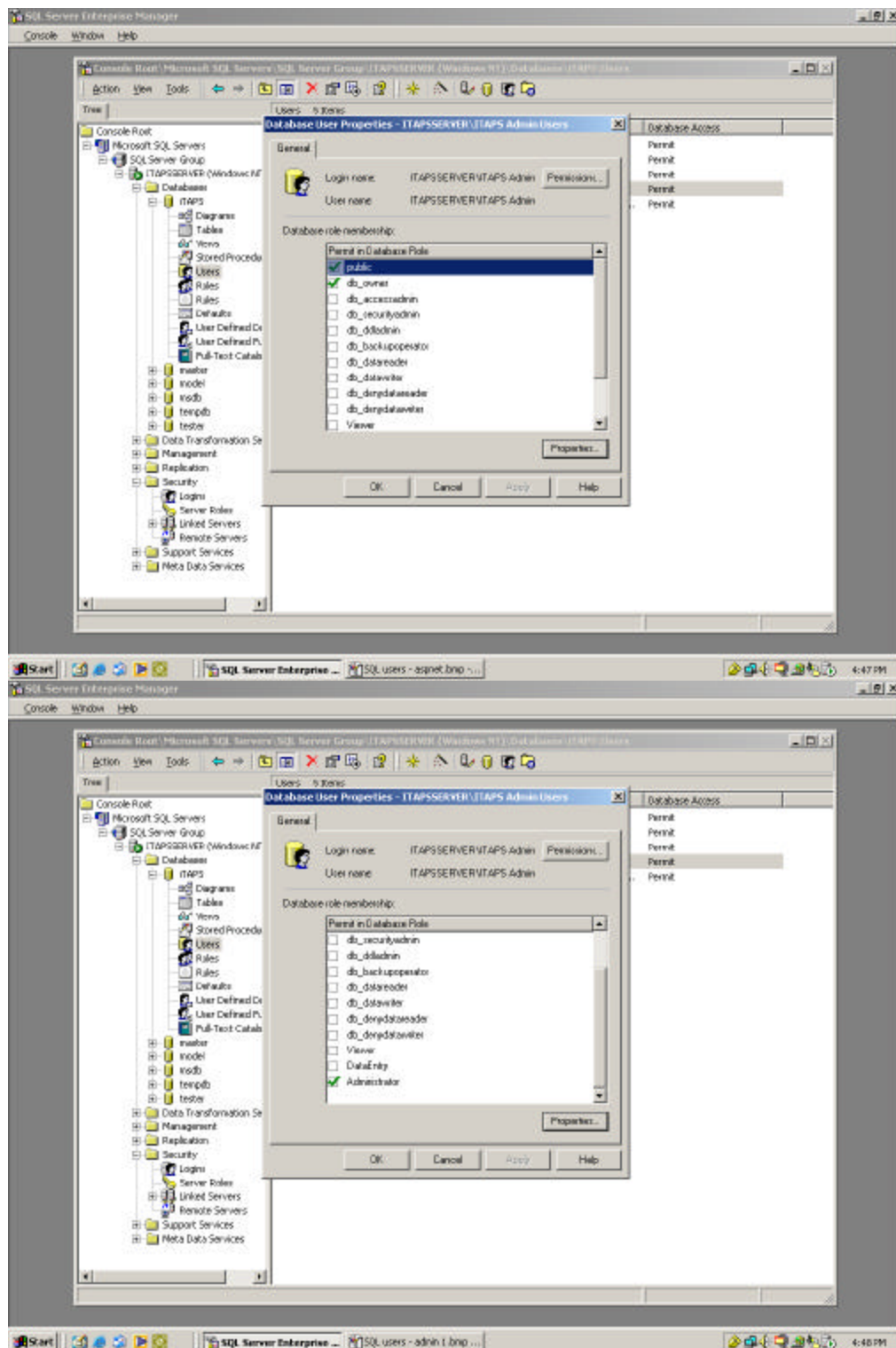
The “dbo” user should have the following roles:

- public
- db_owner



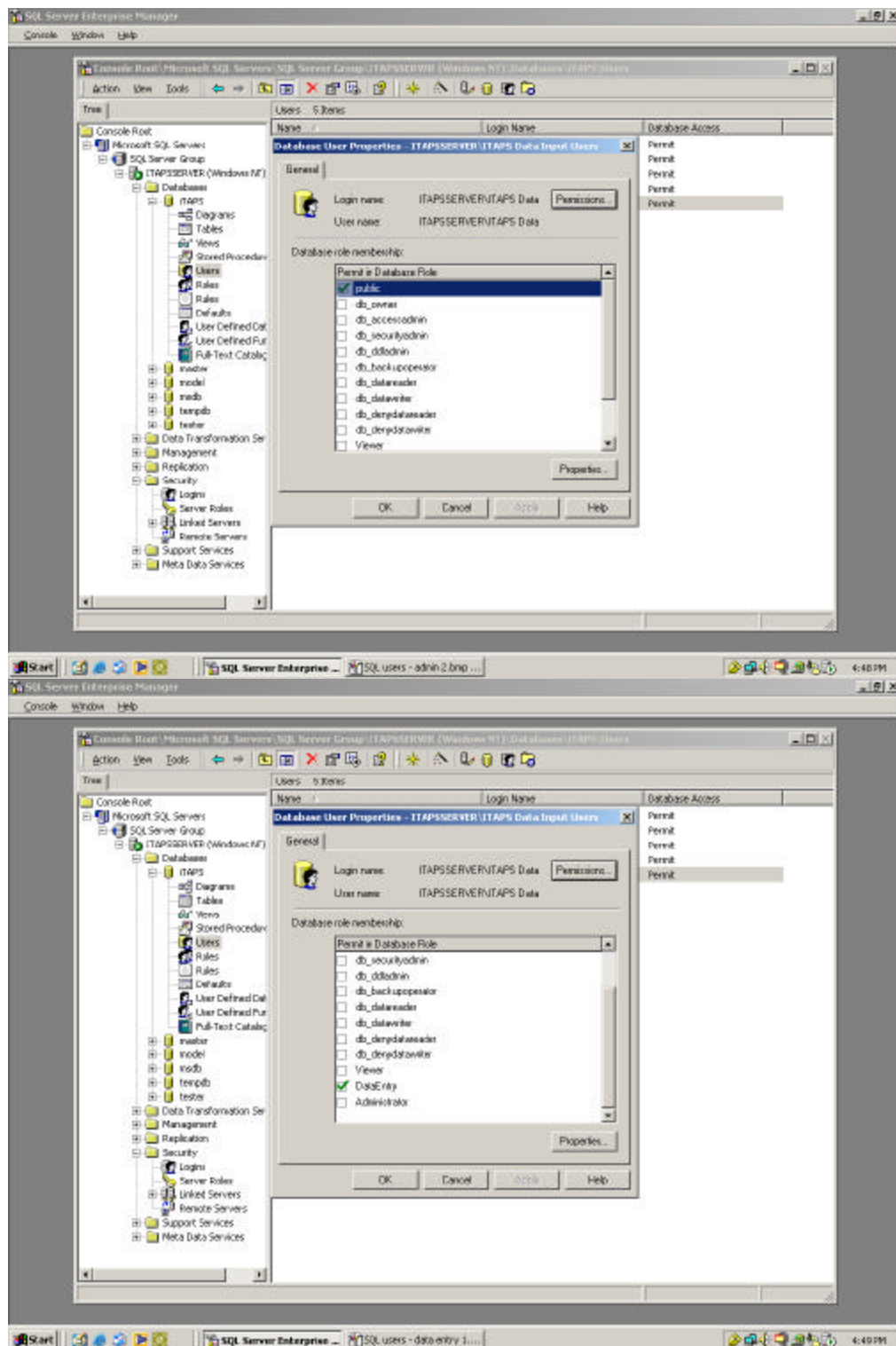
The “ASPNET” user should have the following roles:

- public
- db_owner



The "ITAPS Admin User" user should have the following roles:

- public
- db_owner
- Administrator



The "ITAPS Admin User" user should have the following roles:

- public
- Data Entry

5.0 System Operation

NOTE: There are instructions at the top of each of the individual web forms that instruct the user on how to utilize that pages functions. The user is encouraged to read these instructions in addition to this manual.

5.1 Administrative Users.

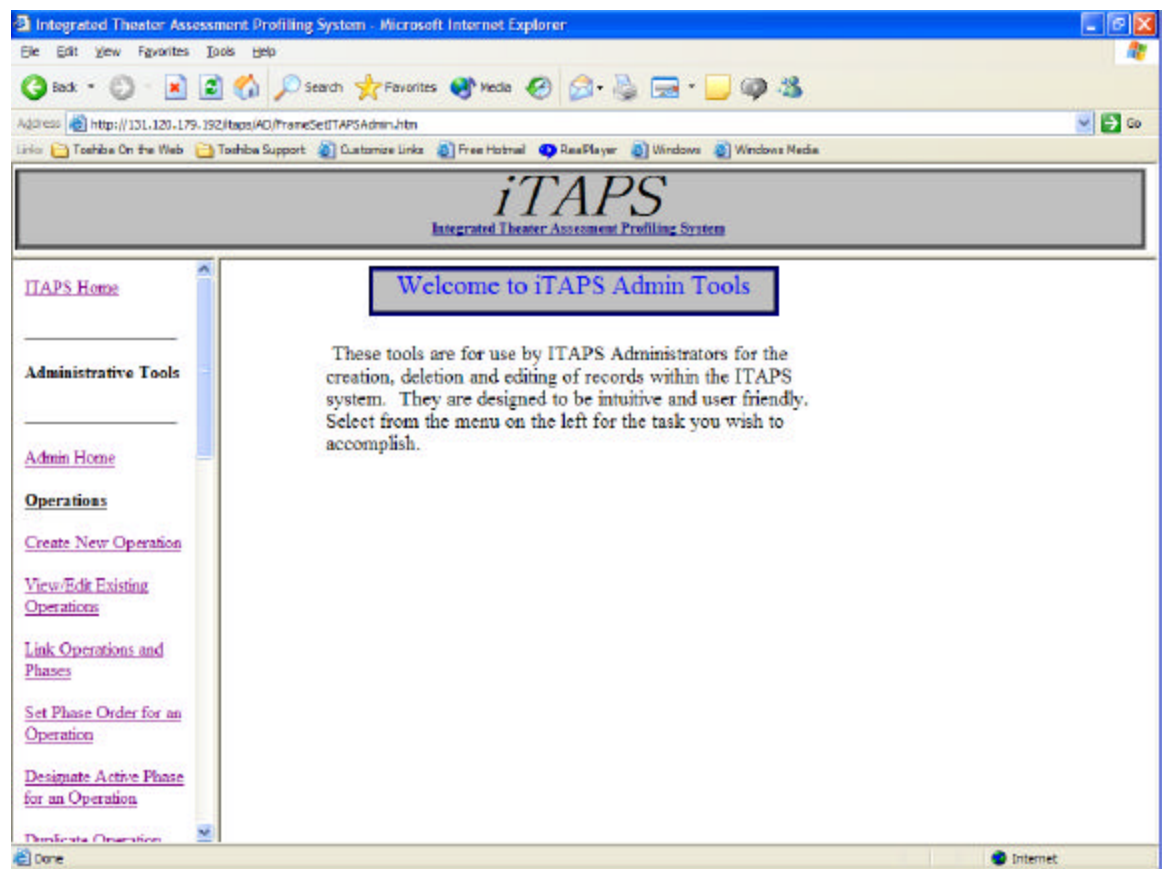


Figure 1. Welcome to Admin Tools

Administrative Users use the Admin Tools to create the structure for the Operations, Phases and Elements (see figure 1). The following sections detail the accomplishment of all admin functions.

5.1.1 Creating Operations

The screenshot shows a web browser window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows the URL "http://131.129.179.192/itaps/AD/FrameSetITAPSAdmin.htm". The page features a header with the "iTAPS" logo and the text "Integrated Theater Assessment Profiling System". A left sidebar contains a menu with links: "iTAPS Home", "Administrative Tools", "Admin Home", "Operations", "Create New Operation", "View/Edit Existing Operations", "Link Operations and Phases", "Set Phase Order for an Operation", "Designate Active Phase for an Operation", and "Designate Operation". The main content area is titled "Create New Operation" and contains the following form elements:

- Instructions:** Enter a new Operation Title, Description and active status, then press the "Submit" button. iTAPS will ask you to confirm your entry. Press "Continue" if the entry
- New Operation Title:** Text input field containing "Test Operation Entry".
- Operation Description:** Text area containing "Test Operation Description".
- Will Operation be Active?:** Radio button control with "No" and "Yes" options. The "Yes" option is selected.
- Submit:** A button to submit the form.

Figure 2. Creating an Operation

Administrative Users create Operations by selecting "Create New Operation" from the Admin Tools menu (see figure 2). The Operation Title is mandatory, the Operation Description is optional. A radio button control is provided to indicate whether an Operation is active or not. The Active status flag is designed to help the user segregate Operations currently in progress and old Operations or Operations being created. Once your entry is correct, click the "Submit" button.

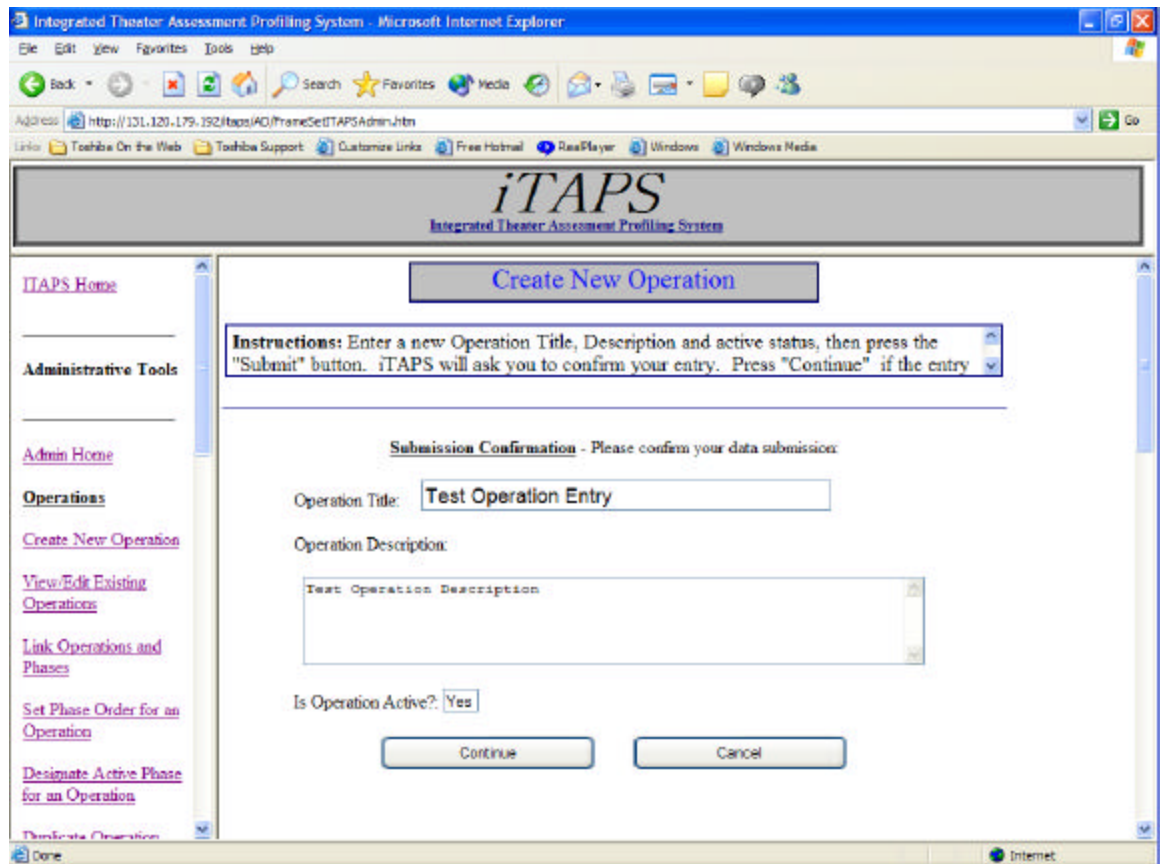


Figure 3. Confirming your entry

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 3). If you made a mistake or do not wish to create this Operation entry, Click “Cancel”. You will be returned to the previous screen. If the entry is correct, click “Continue” and ITAPS will update its database.

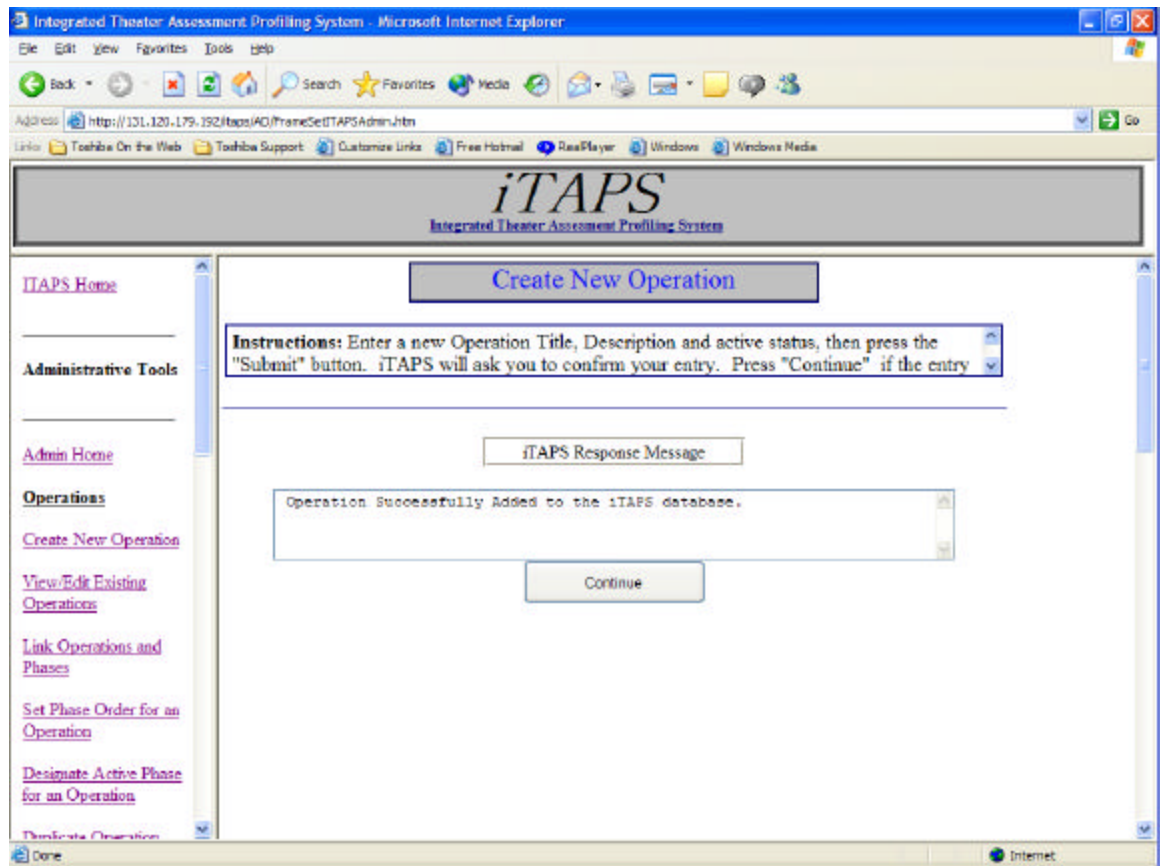


Figure 4. iTAPS responds

iTAPS will display a response screen concerning your submission (see figure 4). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.2 Editing/Deleting Operations

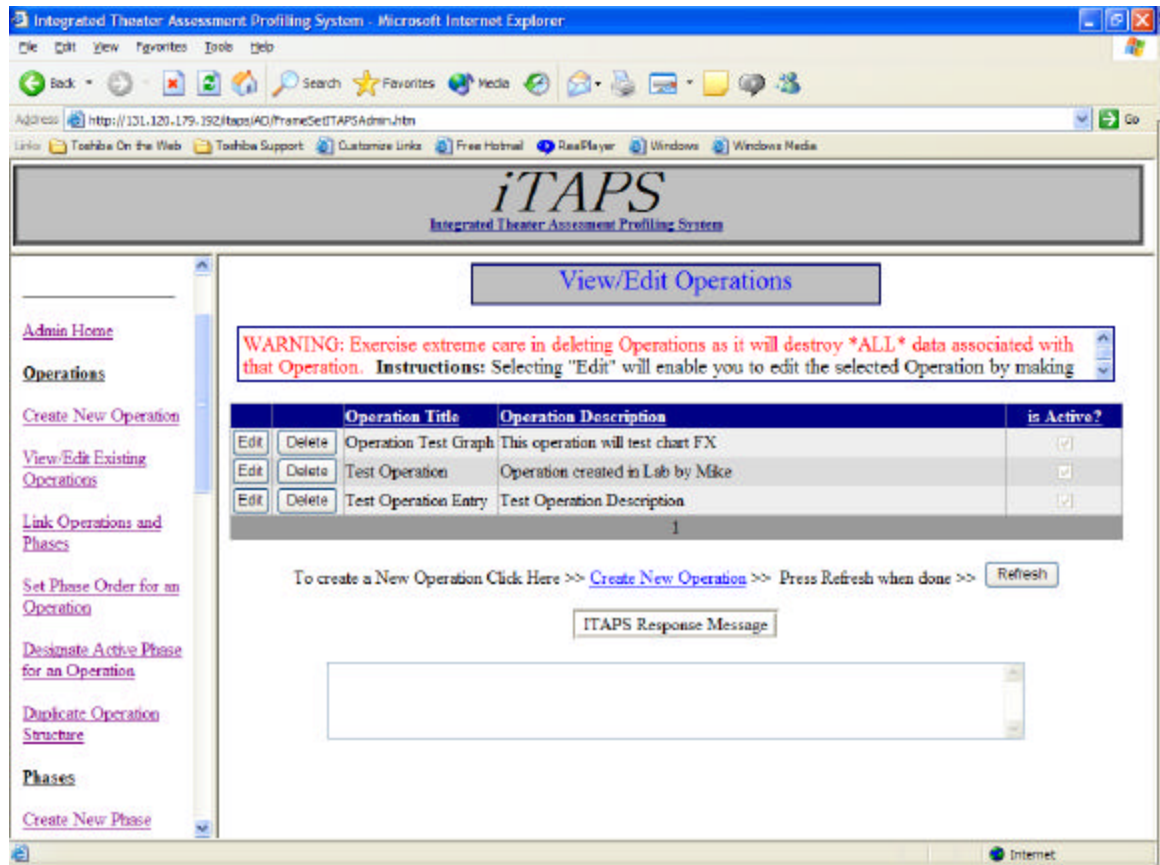


Figure 5. Viewing and Editing Operations

To view, edit or delete Operations already in ITAPS select "View/Edit Operations" from the Admin Tools menu (see figure 5). The Operation information is displayed in a tabular grid format. The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Operation click the "Create New Operation" hyperlink. After creating the new Operation click "Refresh" to update the datagrid with your entry.

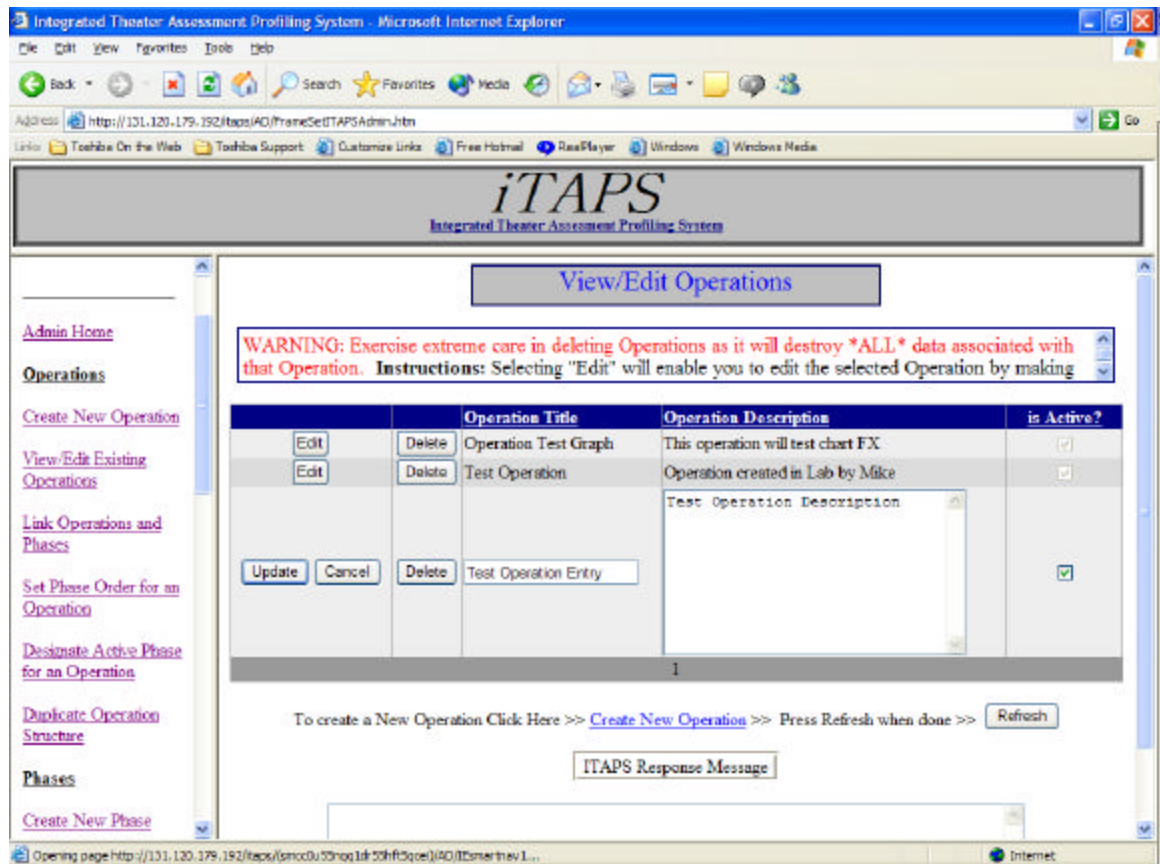


Figure 6. Editing an Operation

To Edit an Operation Click the “Edit” button for that Operation. An Edit Template will be displayed for that Operation (see figure 6). Make the changes you desire and click “Update” to enter your changes or “Cancel” to cancel your changes.

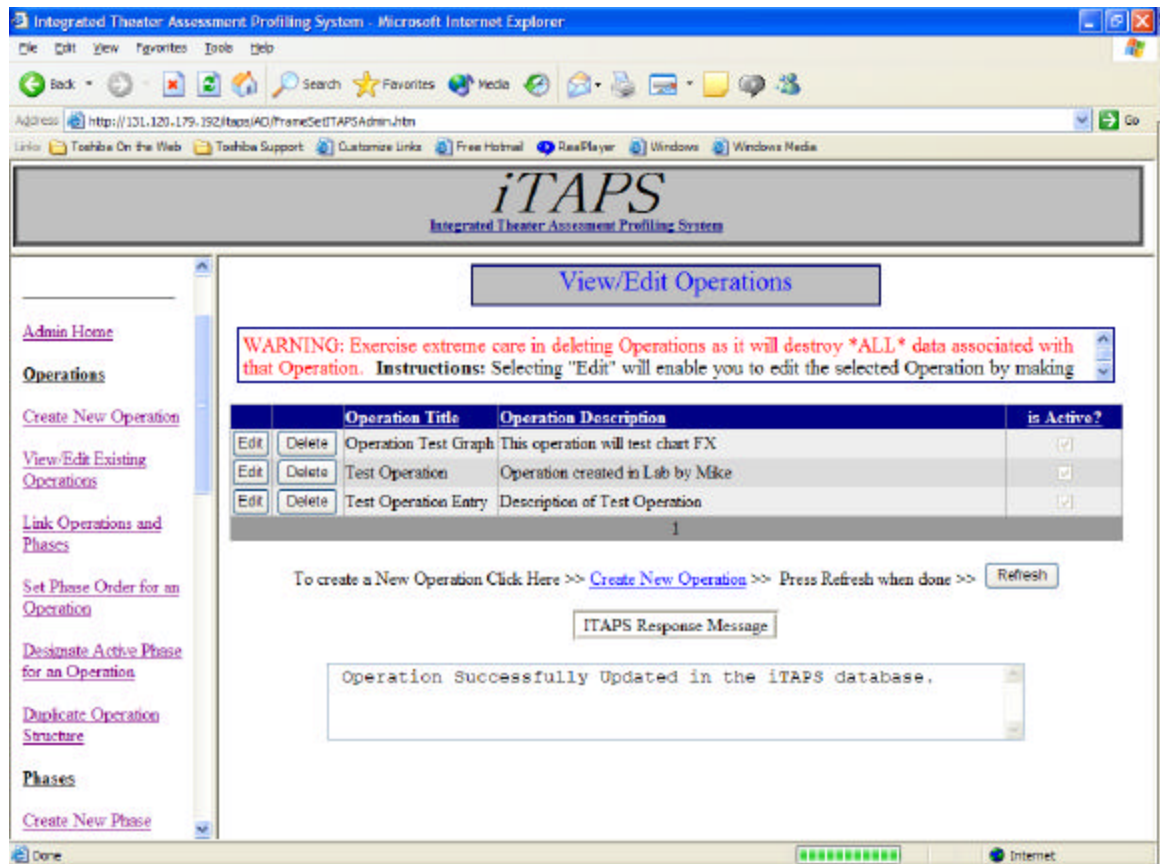


Figure 7. ITAPS responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 7). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

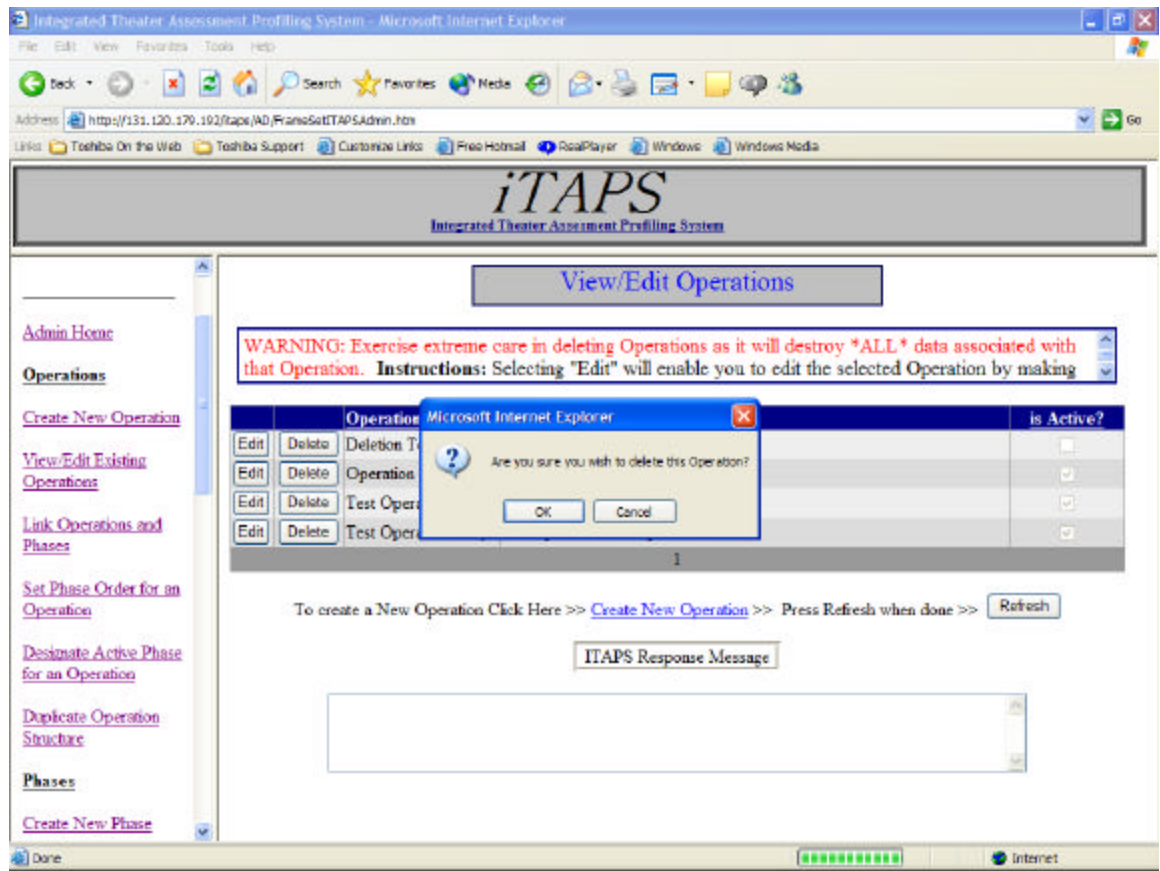


Figure 8. Deleting an Operation

To Delete an Operation, click the “Delete” button for the desired Operation. A confirmation box will ask you to confirm your selection (see figure 8). Clicking “OK” will cause the Operation and all associated data to be permanently erased. Clicking “Cancel” will cancel your delete request. **WARNING: Exercise extreme care in deleting Operations as it will destroy *ALL* data associated with that Operation.**

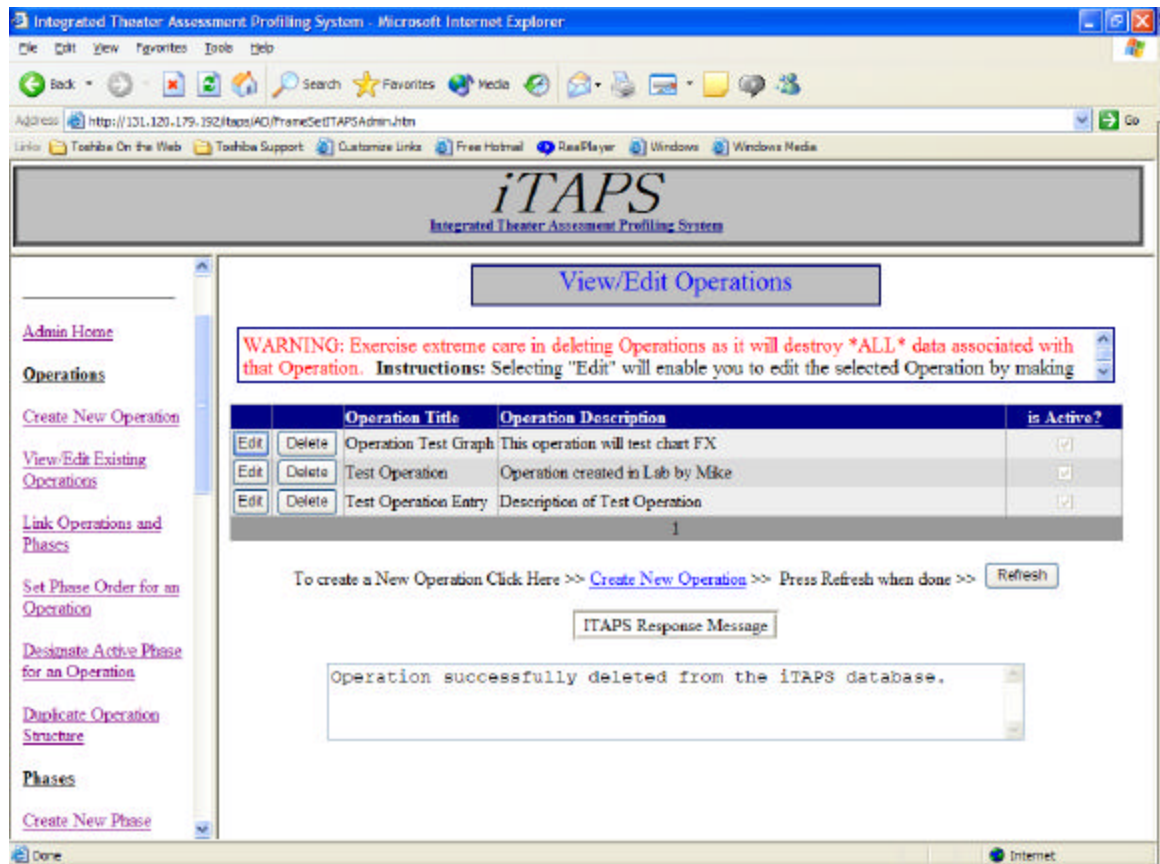


Figure 9. ITAPS responds

After Deletion, ITAPS will list the results of the operation in the Response Message window (see figure 9). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.3 Creating Phases

The screenshot shows a web browser window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.129.179.192/itaps/AD/PhaseSet/ITAPSAdmin.htm". The page features a navigation menu on the left with links like "Operations", "Link Operations and Phases", "Set Phase Order for an Operation", "Designate Active Phase for an Operation", "Duplicate Operation Structure", "Phases", "Create New Phase", "View/Edit Existing Phases", "Link Phases and Operations", and "Set Phase Order for an Operation". The main content area is titled "Create New Phase" and contains an instruction box: "Instructions: Enter a new Phase Title and Description and which Operation to link this phase to. Press 'Submit' once you make your entries. iTAPS will then ask you to confirm". Below this, there are input fields for "New Phase Title" (containing "Phase Test Entry 1"), "Phase Description" (containing "Phase Test Entry Description"), "Linked to Operation" (a dropdown menu set to "-- None --"), "Operations List Filter" (radio buttons for "All" and "Active", with "Active" selected), and "Phase Order" (a dropdown menu). A "Submit" button is located at the bottom right of the form.

Figure 10. Creating a Phase

Administrative Users create Operations by selecting "Create New Operation" from the Admin Tools menu (see figure 10). Phase Title is mandatory, all other entries are optional.

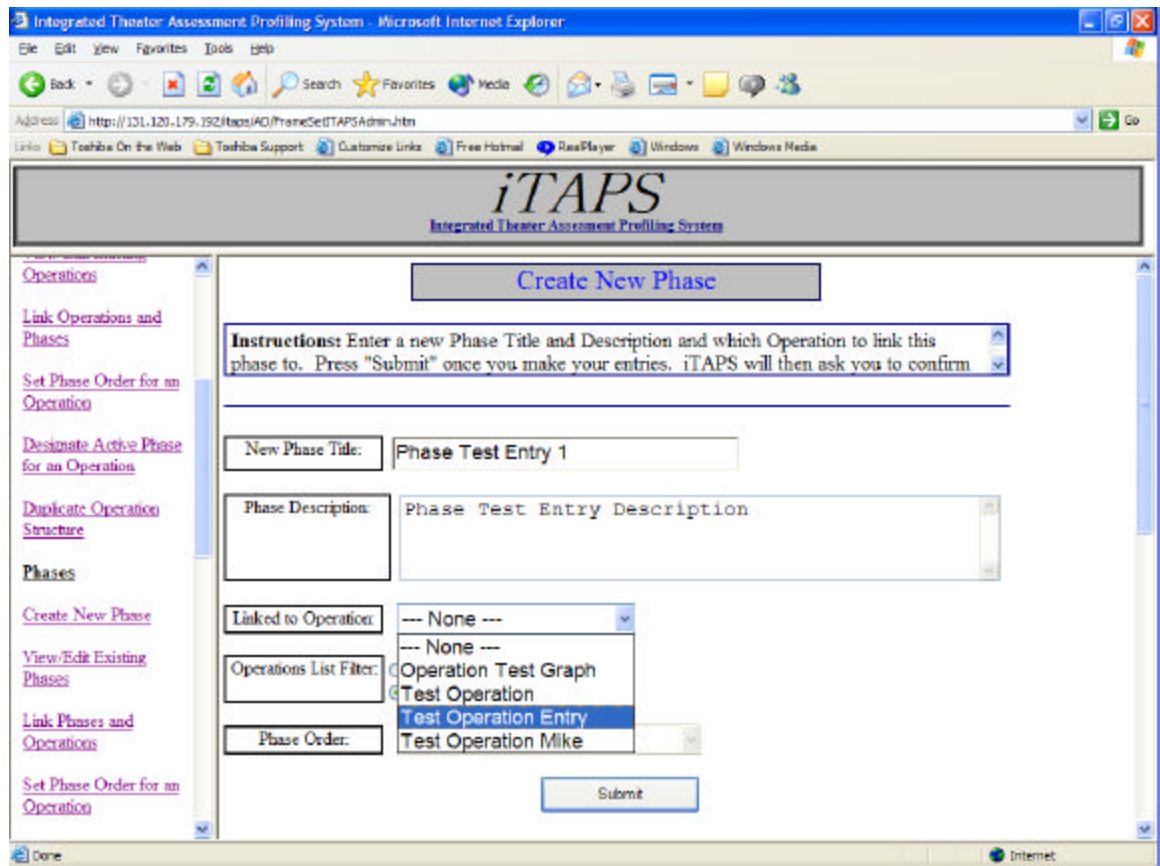


Figure 11. Selecting an Operation

To Select an Operation to link this Phase to, use the "Operation List Filter" radio button to select whether to include all Operations or only active Operations in the "Linked to Operation" drop down list. Select the Operation you wish to link to this phase with the drop down list (see figure 11).

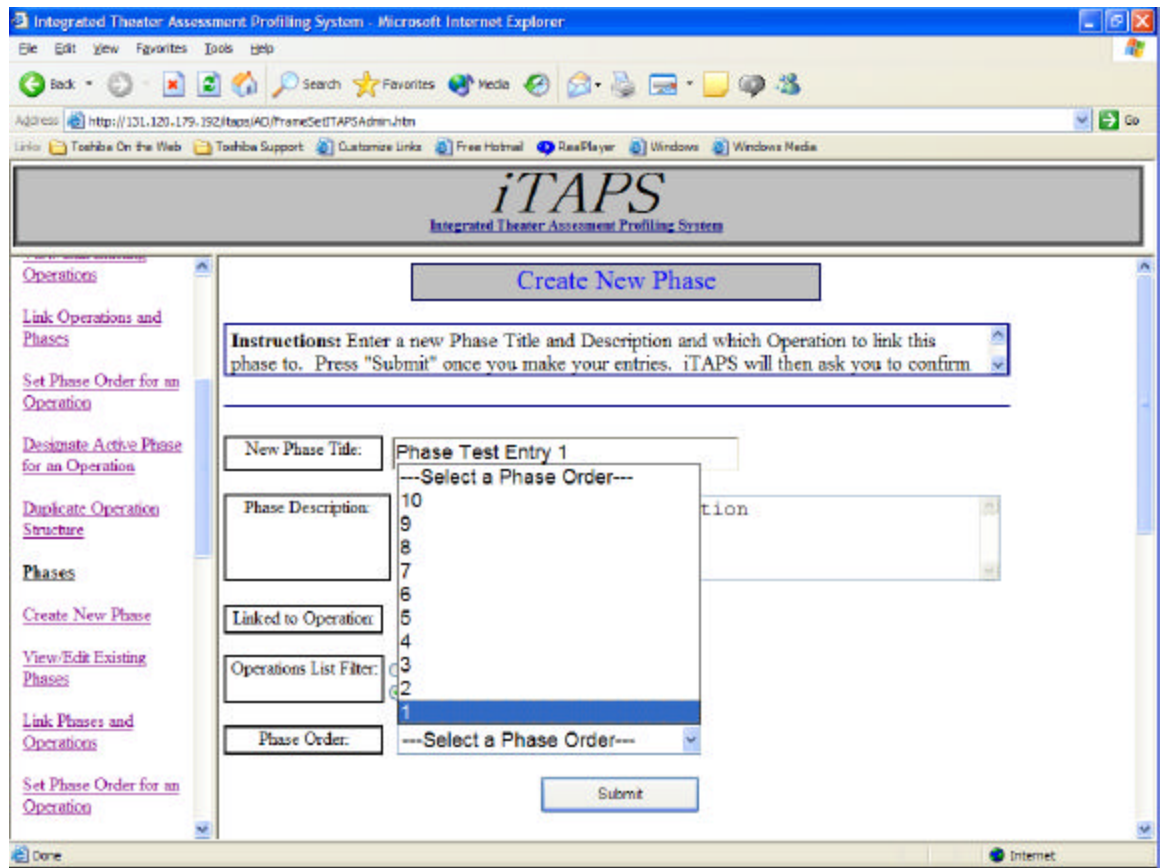


Figure 12. Selecting a Phase Order

The Phase Order drop down list now becomes available, listing the phase order slots still available for this Operation (see figure 12). **Each Operation is limited to 10 Phases.** If an Operation link is chosen, the Phase Order selection becomes mandatory. Once your entry is correct, click the "Submit" button.

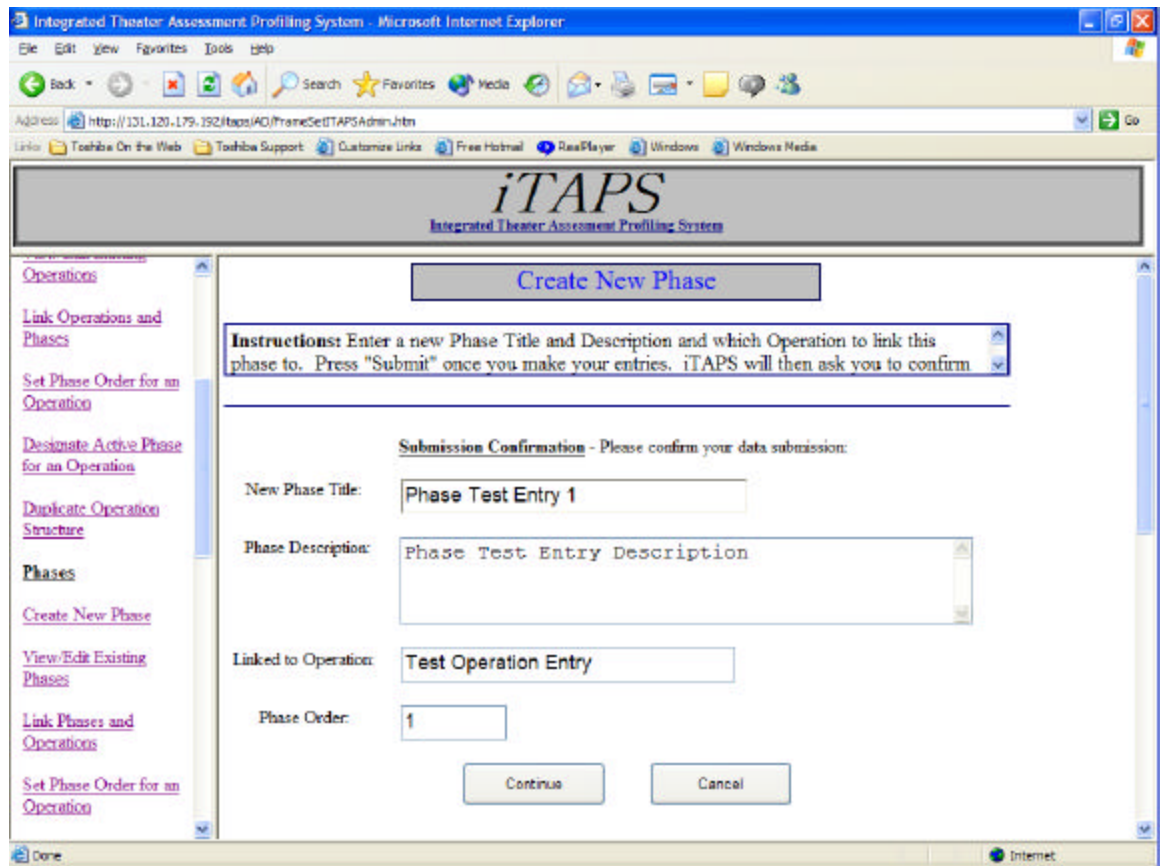


Figure 13. Confirming your entry

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 13). If you made a mistake or do not wish to create this Phase entry, Click "Cancel". You will be returned to the previous screen. If the entry is correct, click "Continue" and ITAPS will update its database.

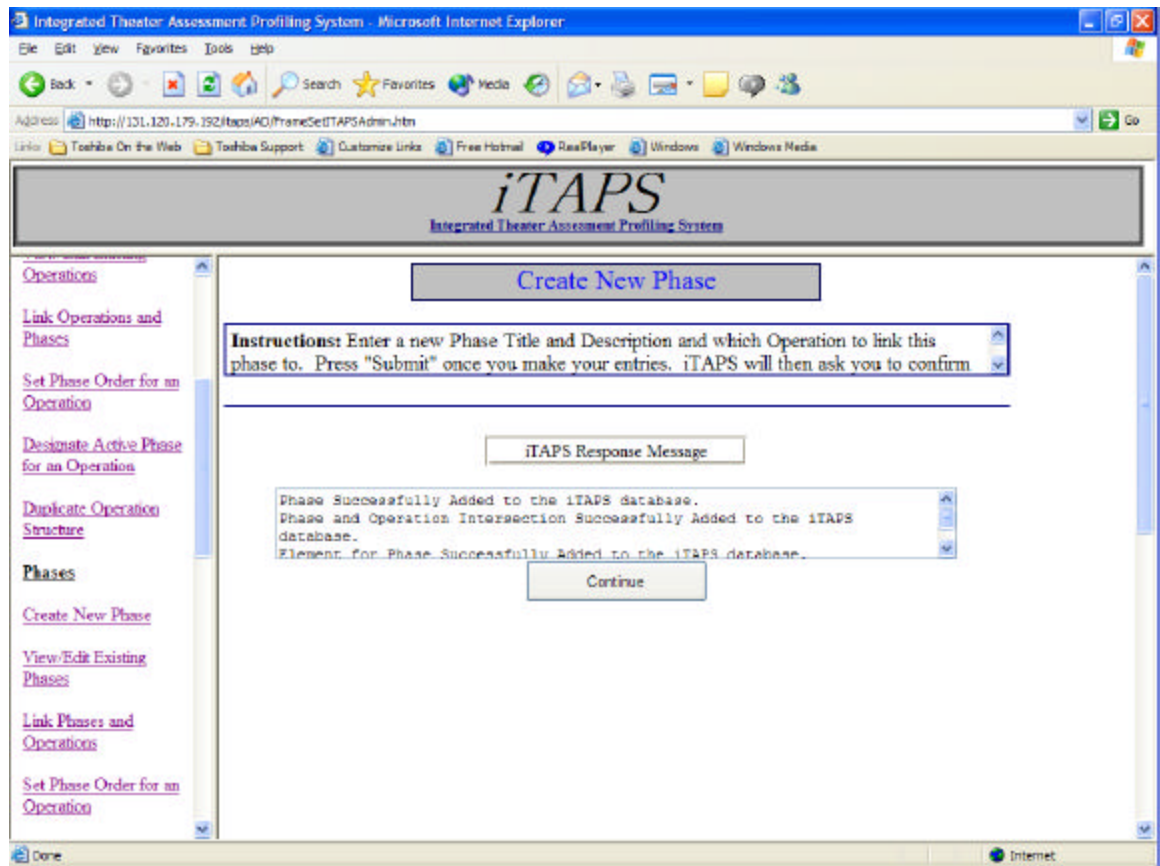


Figure 14. iTAPS responds

iTAPS will display a response screen concerning your submission (see figure 14). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.4 Editing/Deleting Phases

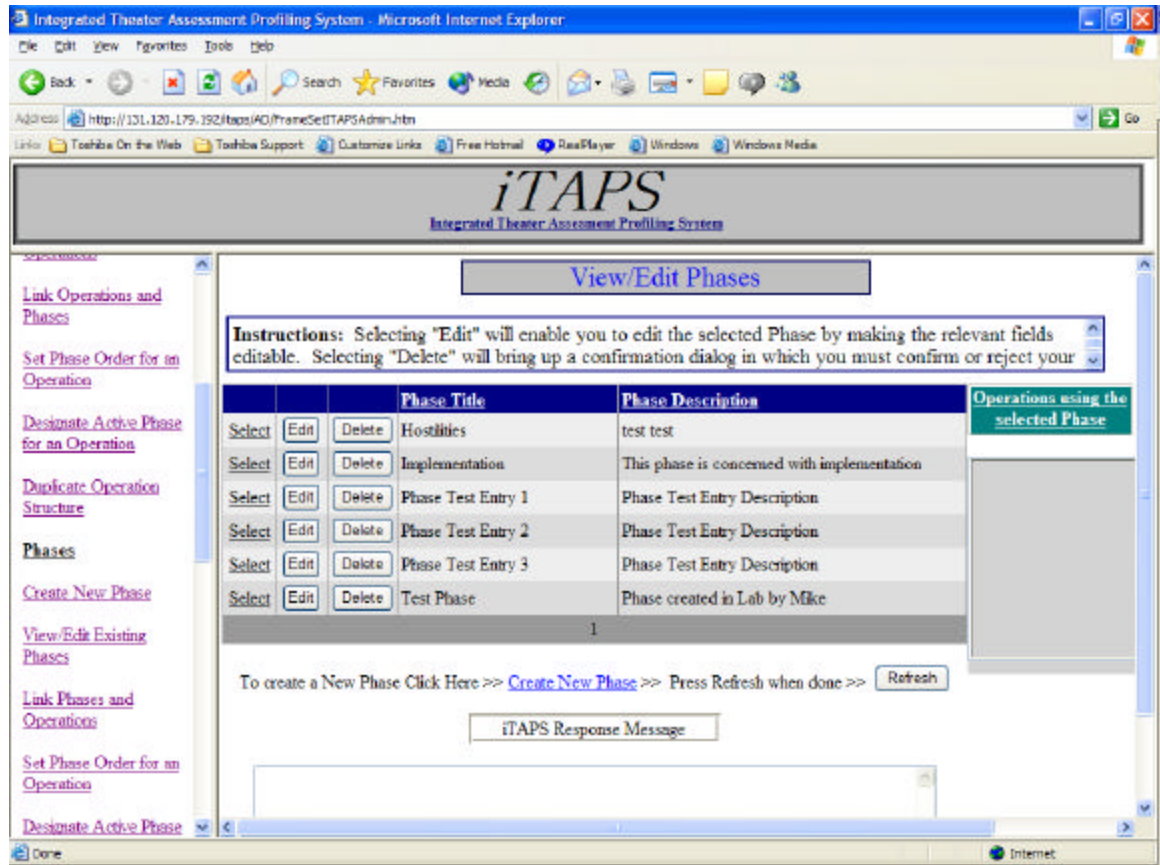


Figure 15. Viewing and Editing Phases

To view, edit or delete Phases already in ITAPS select "View/Edit Phases" from the Admin Tools menu (see figure 15). The Phase information is displayed in a tabular grid format. The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header.

Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Phase, Click the "Create New Phase" hyperlink. After creating the new Phase click "Refresh" to update the datagrid with your entry.

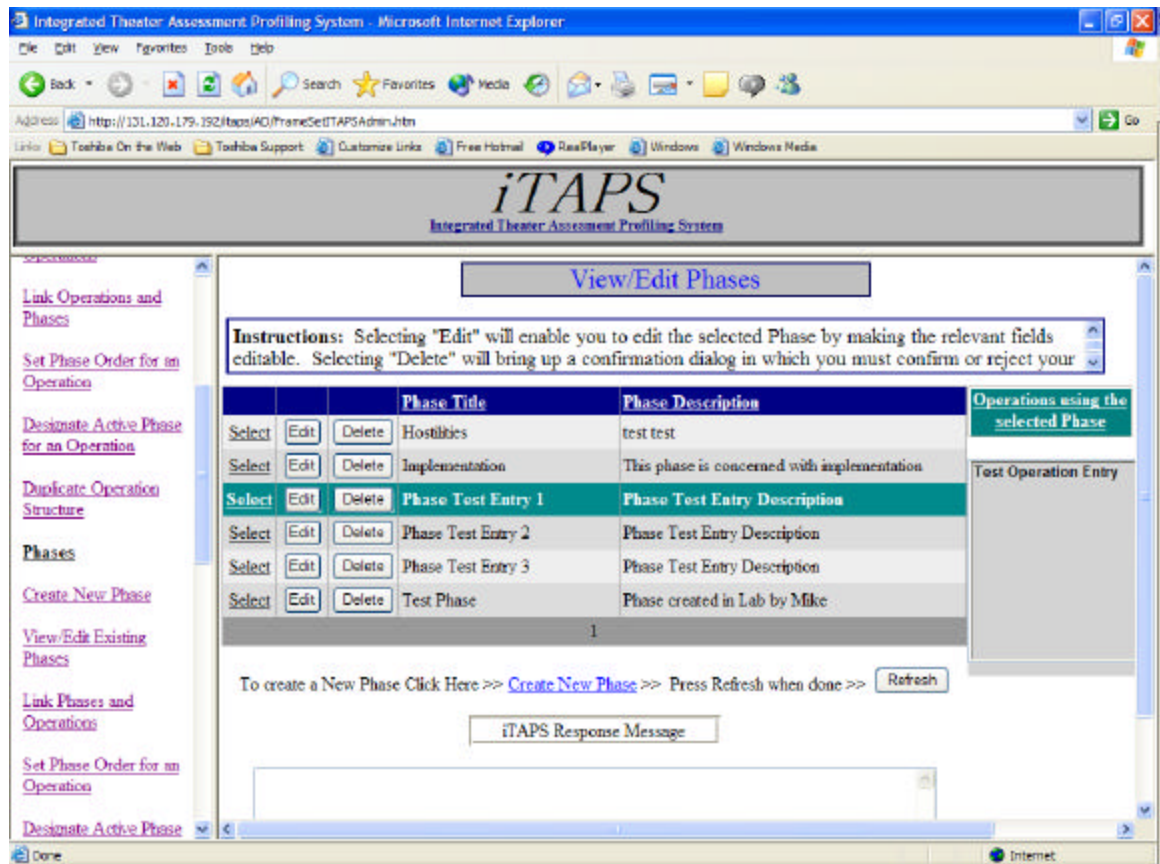


Figure 16. Selecting a Phase

Operations that use this Phase will be listed in the listbox to the right when a particular Phase is selected. Phases that are linked to Operations cannot be deleted. You may remove the Operation/Phase link via the "Link Phases and Operations" web form.

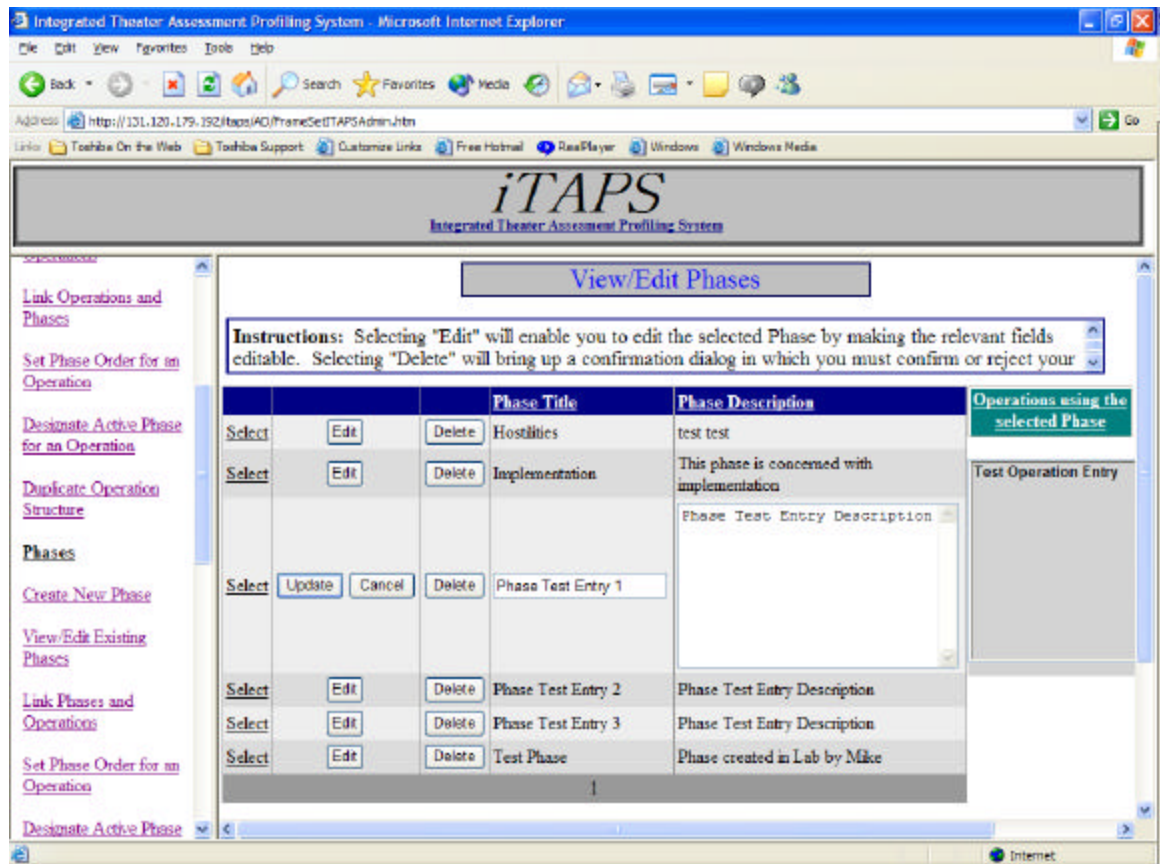


Figure 17. Editing a Phase

To Edit a Phase, click the “Edit” button for that Phase. An Edit Template will be displayed for that Phase (see figure 17). Make the changes you desire and click “Update” to enter your changes or “Cancel” to cancel your changes.

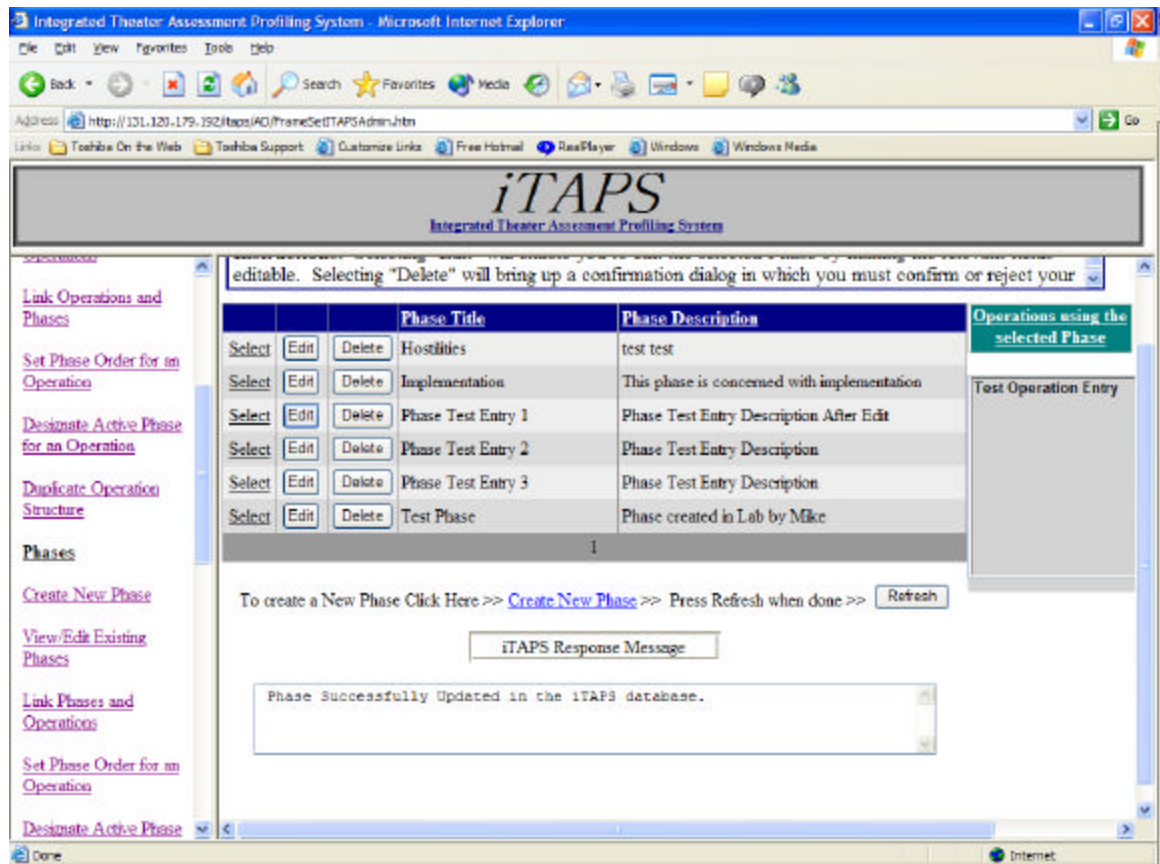


Figure 18. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 18). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

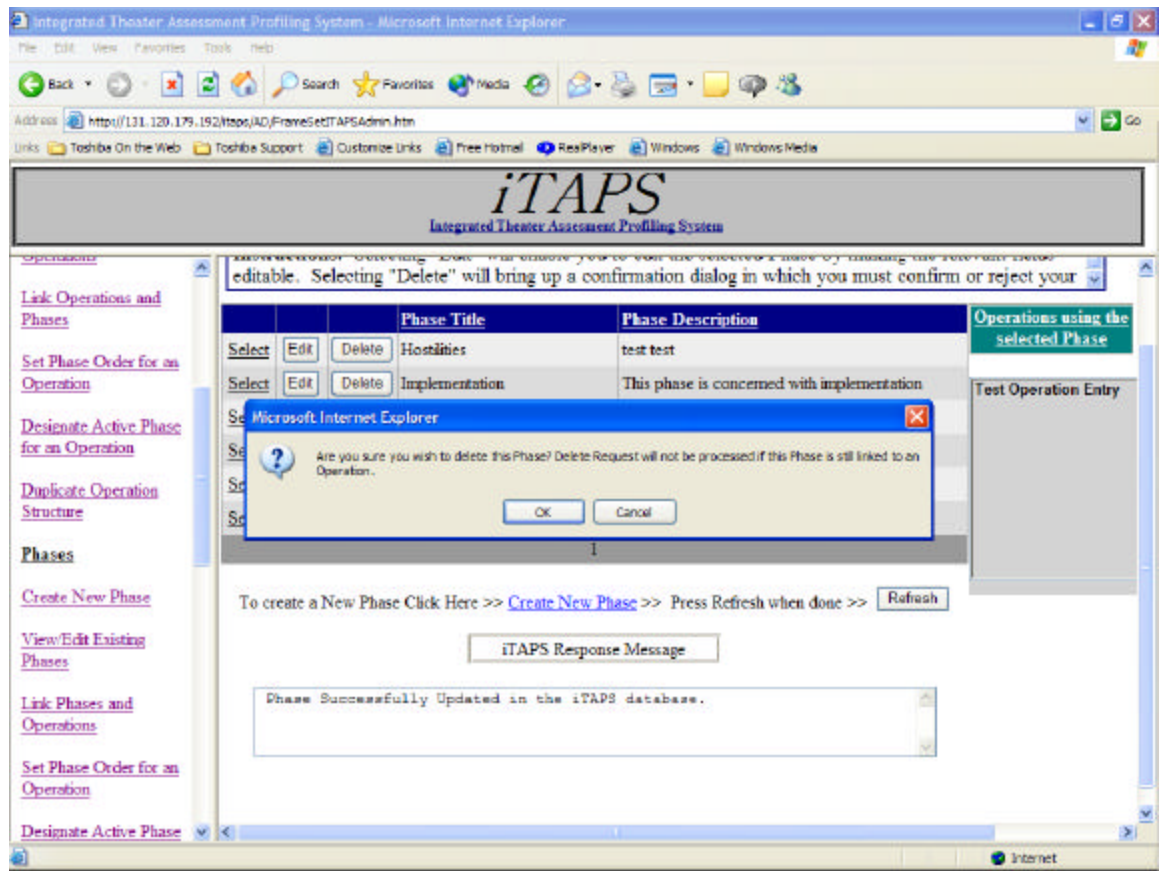


Figure 19. Deleting a Phase

To Delete a Phase, click the “Delete” button for the desired Phase. A confirmation box will ask you to confirm your selection (see figure 19). Clicking “OK” will cause the Phase to be permanently deleted. Clicking “Cancel” will cancel your delete request. Phases that are linked to Operations cannot be deleted. You may remove the Operation/Phase link via the “Link Phases and Operations” web form.

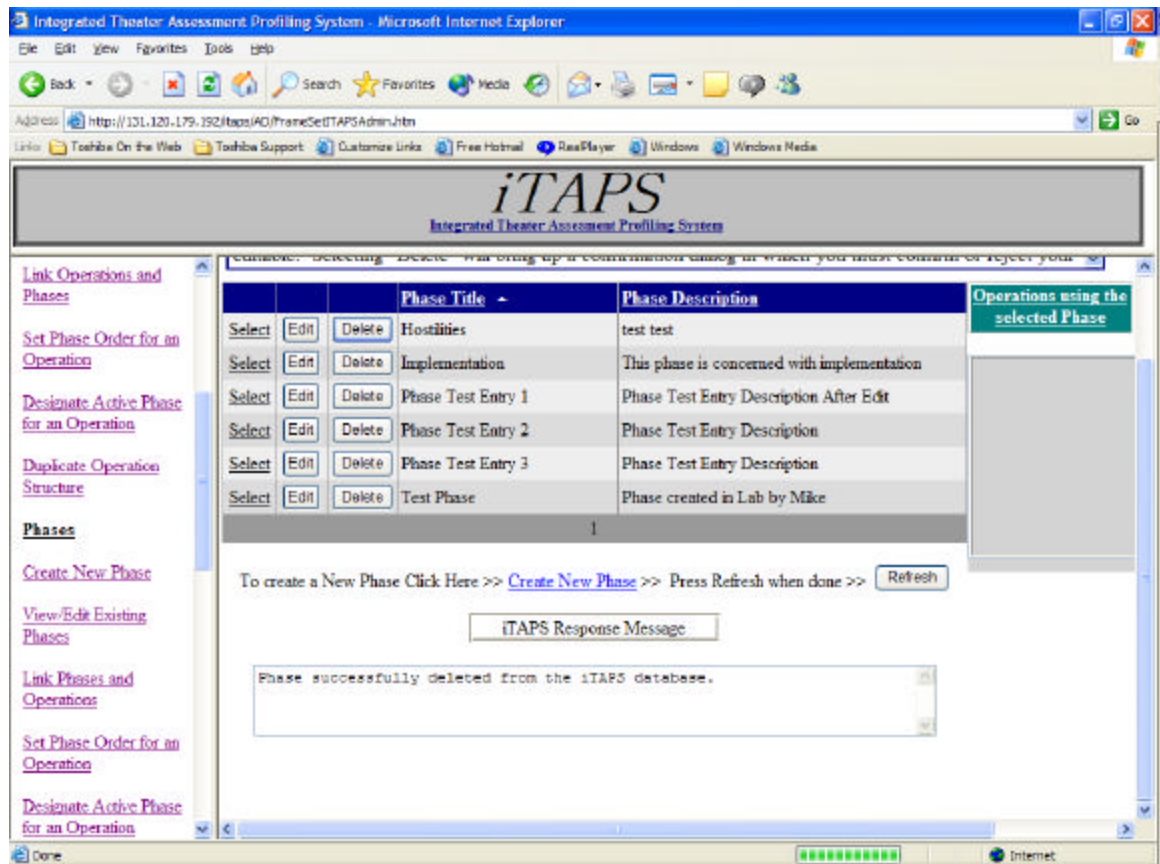


Figure 20. ITAPS Responds

ITAPS will display a response screen concerning your submission (see figure 20). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.5 Creating Elements

The screenshot shows a web browser window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.129.179.192/itaps/AD/FrameSetITAPSAdmin.htm". The page has a header with the "iTAPS" logo and the text "Integrated Theater Assessment Profiling System". On the left is a navigation menu with links: "View/Edit Existing Phases", "Link Phases and Operations", "Set Phase Order for an Operation", "Designate Active Phase for an Operation", "Elements" (selected), "Create New Element" (highlighted), "View/Edit Existing Elements", "Create New Element Title", "View/Edit Element Titles", and "Graders". The main content area is titled "Create New Element". It contains an "Instructions" box stating: "Enter New Element data below. When complete, press 'Submit'. iTAPS will ask you to confirm your entries. Press 'Continue' if the entries are correct, press". Below this is an "IMPORTANT" box: "Element MUST be linked to an Operation, Phase and Parent Element". It includes links "Create Operation" and "Create Phase", and a "Refresh" button. The form fields are: "Select Operation:" with a dropdown menu showing "--Select an Operation--"; a note "(Please wait a moment after your Operation selection for the 'Phase' drop down list to fill)"; "Operations List Filter:" with radio buttons for "All" and "Active"; "Select Phase:" with a dropdown menu; "Select a Side Color:" with a dropdown menu showing "Blue"; and "Select Parent Element:" with a dropdown menu. A note "(Please wait a moment after your Phase selection for the 'Parent Element' drop down list to fill)" is below the last two fields.

Figure 21. Creating a new Element

Administrative Users create Elements by selecting "Create New Element" from the Admin Tools menu (see figure 21). All fields are mandatory except for Element Description and Grader.

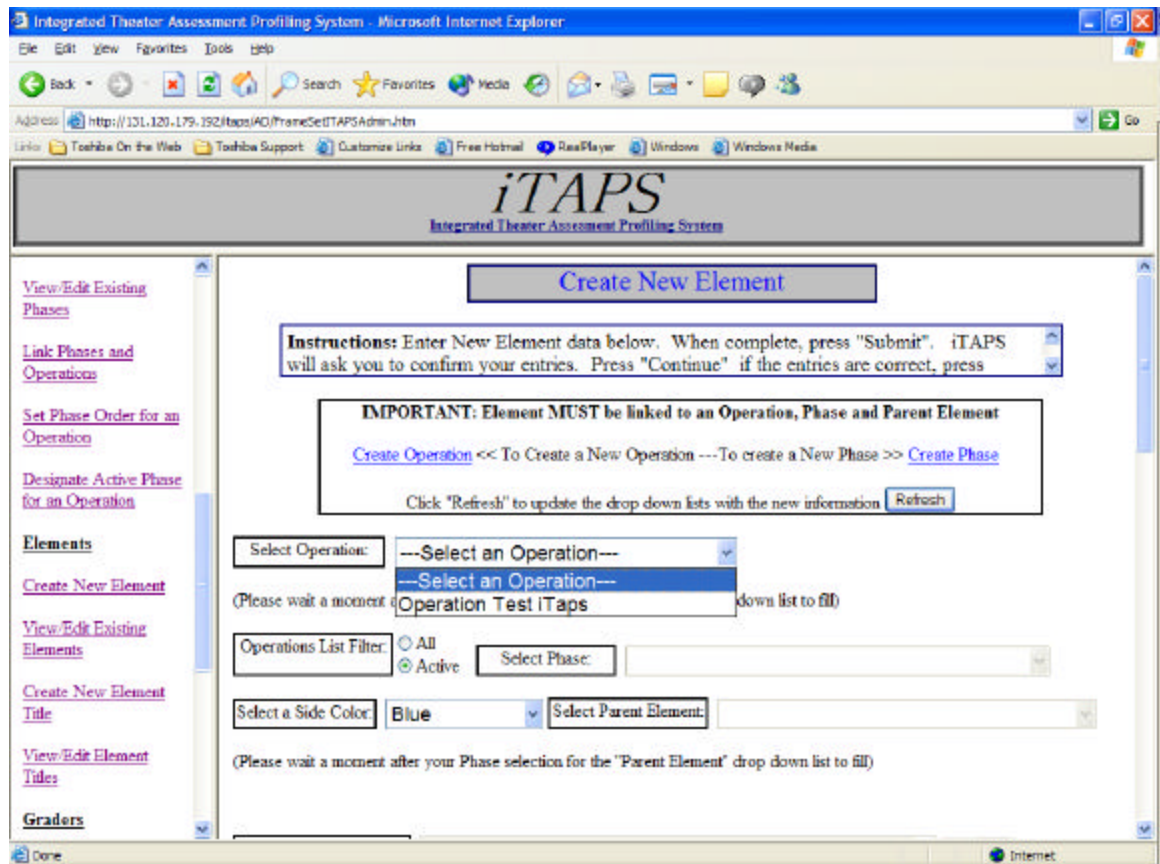


Figure 22. Selecting an Operation

To select an Operation to link this Element to, use the Operation drop down list and the Operation List Filter. The Operation List Filter helps the user narrow or expand the list of Operations in the Operation drop down list. Once an Operation is selected, ITAPS searches its database for the Phases belonging to that Operation and fills the Phases drop down list. After selecting an Operation, wait a moment for the Phases drop down list to fill. This entry is mandatory.

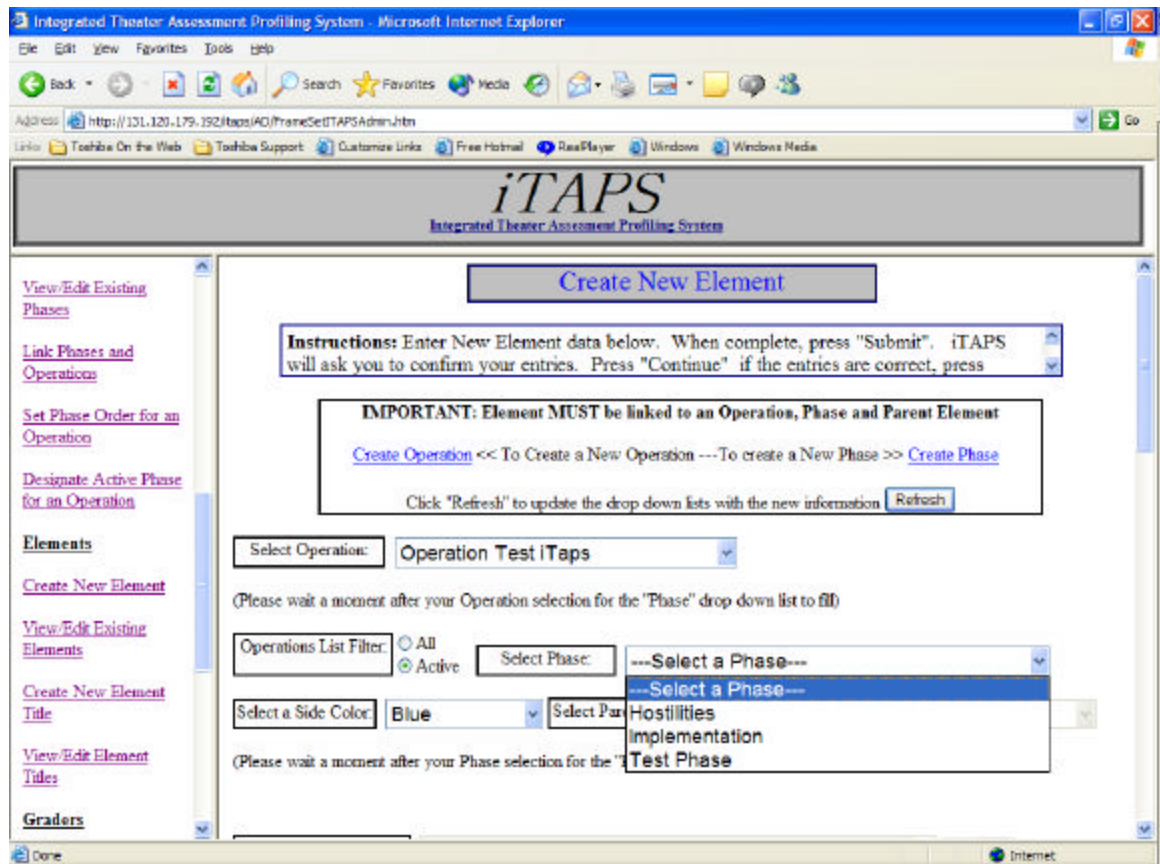


Figure 23. Selecting a Phase

To select a Phase to link this Element to, use the Phase drop down list. Once you make a Phase selection, ITAPS searches it's database for all Elements belonging to the selected Phase and Operation and fills the Parent Element drop down list. After selecting the Phase, wait a moment for the Parent Element drop down list to fill. This entry is mandatory.

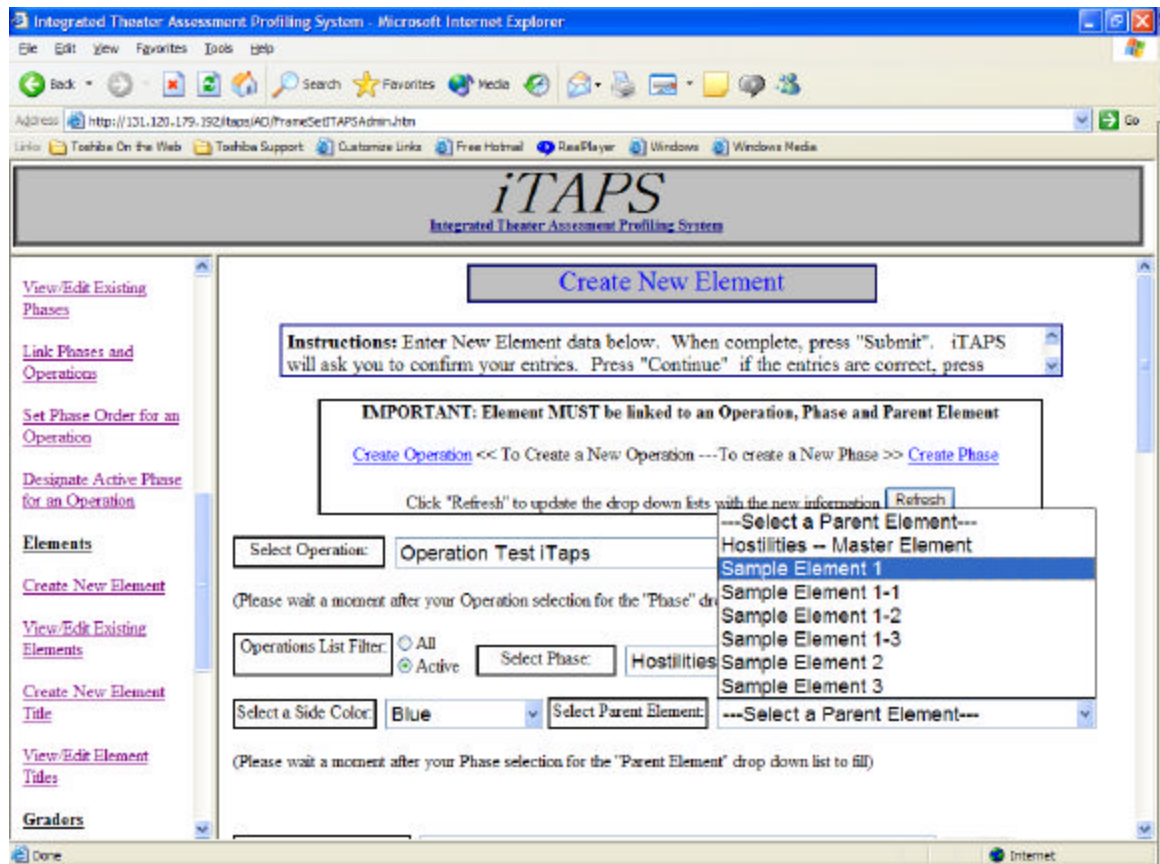


Figure 24. Selecting a Parent Element

The Parent Element may be an Element of the Operation/Phase or it may be the “Master Element” which is the Phase itself. An Element that has the Phase as it’s Parent Element is a top element for that Phase. This entry is mandatory.

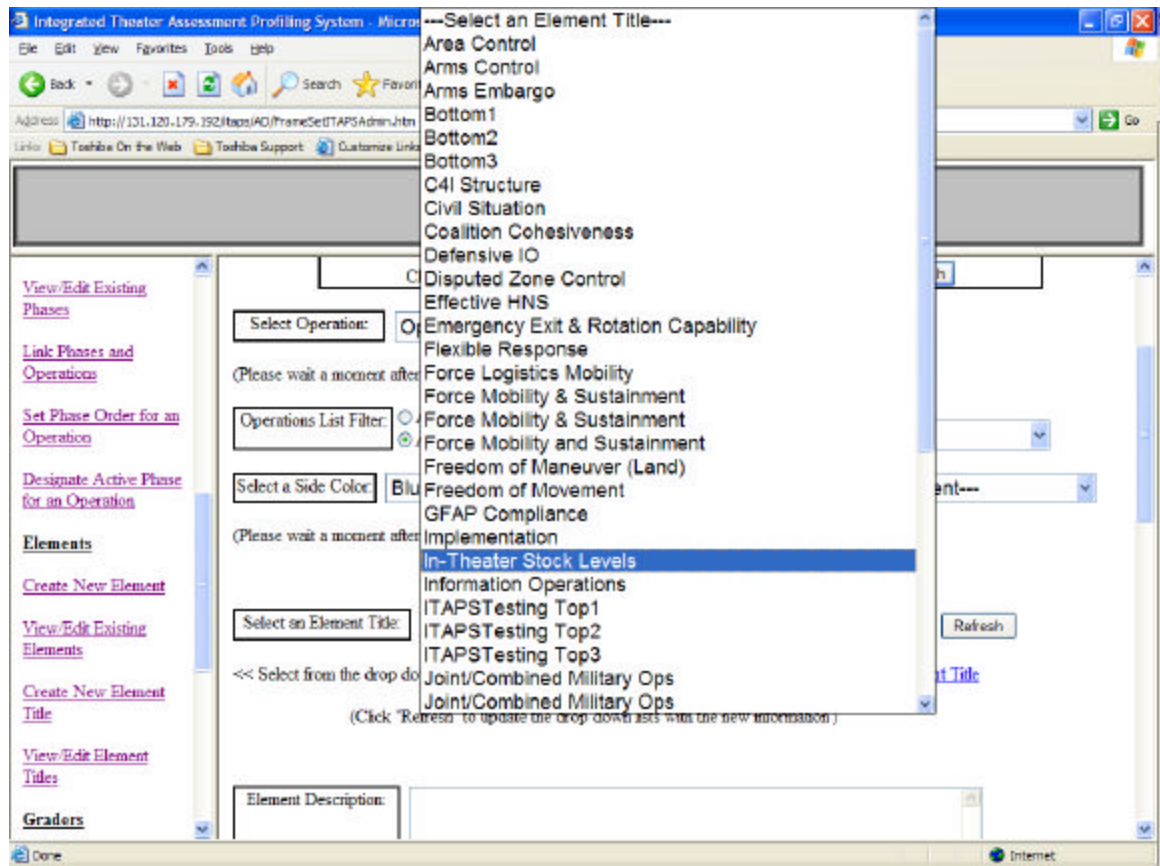


Figure 25. Selecting an Element Title

To select an Element Title, choose one from the Element Title drop down list. This entry is mandatory.

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print Mail

Address http://131.120.179.192/itaps/AG/FrameSetITAPSAdmin.htm Go

Links Toshiba On the Web Toshiba Support Customize Links Free Hotmail RealPlayer Windows Windows Media

iTAPS
Integrated Theater Assessment Profiling System

[View/Edit Existing Phases](#)
[Link Phases and Operations](#)
[Set Phase Order for an Operation](#)
[Designate Active Phase for an Operation](#)
Elements
[Create New Element](#)
[View/Edit Existing Elements](#)
[Create New Element Title](#)
[View/Edit Element Titles](#)
Graders
[Create New Grader](#)

Element Description:

Select a Side Color: Blue

Assign a Grader: ---Auto Graded---

Select a Grade Weighting Factor: 10

Will Element be Active?: ☐ No ☒ Yes

Figure 26. Creating a new Element

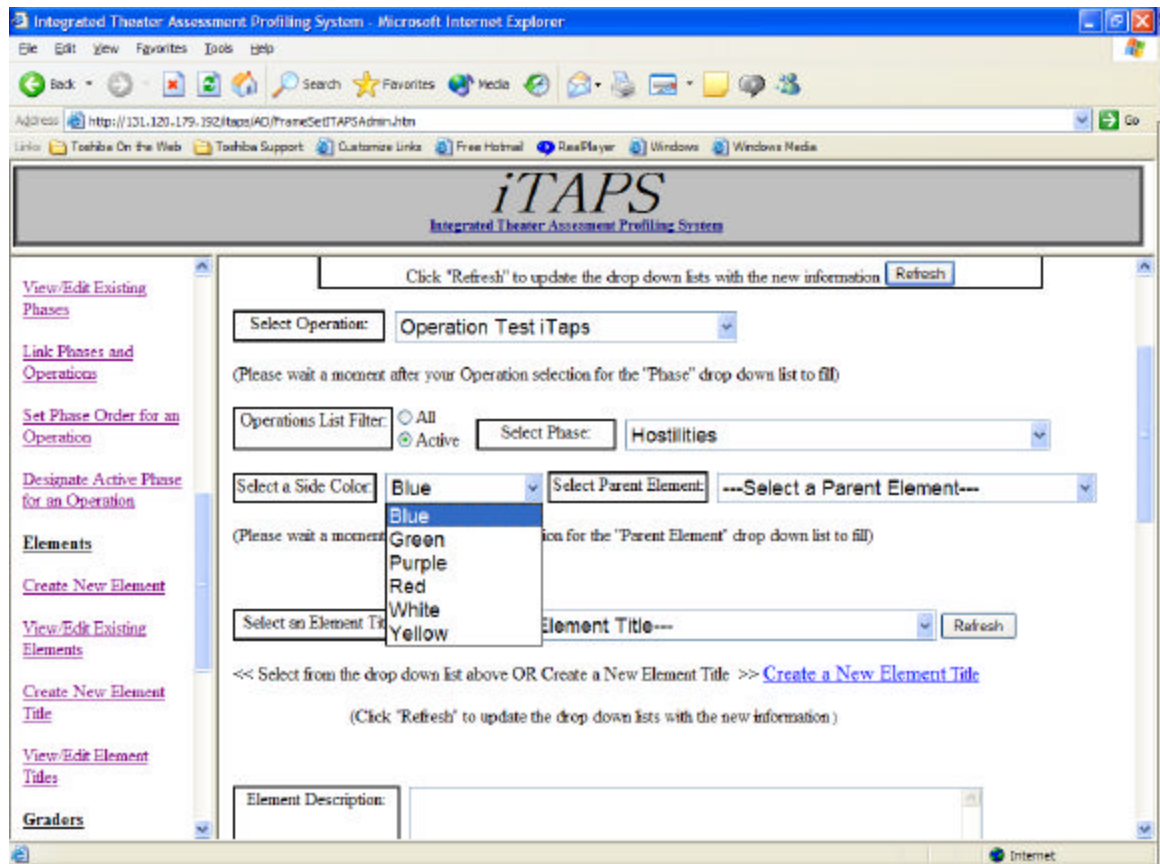


Figure 27. Selecting a Side Color

To select a side color, choose one from the Side Color drop down list.

This entry is mandatory. Note that this selection will change and reload the Parent Element and Element Title drop down lists.

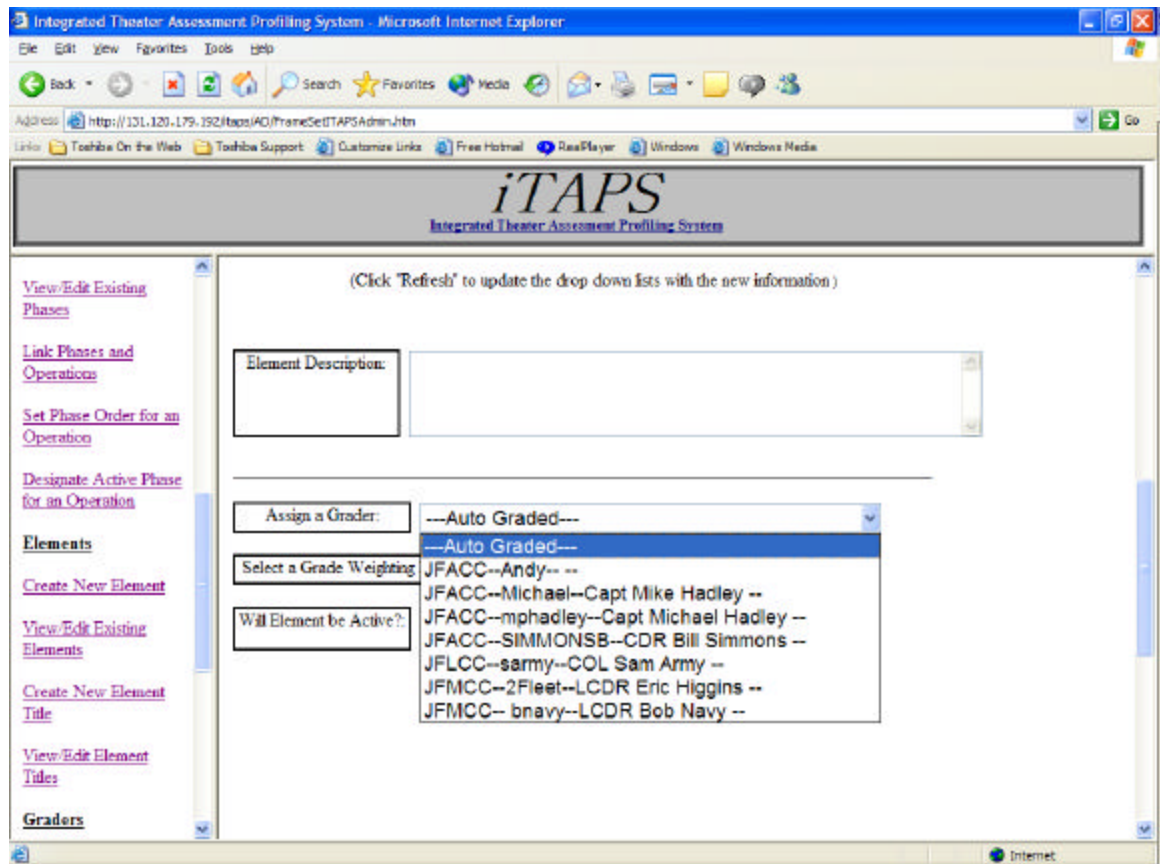


Figure 28. Selecting a Grader

Grader selection is optional. Elements with no specific Grader entry will be autograded by ITAPS based on their child Element grades. To select a grader, use the Grader drop down list.

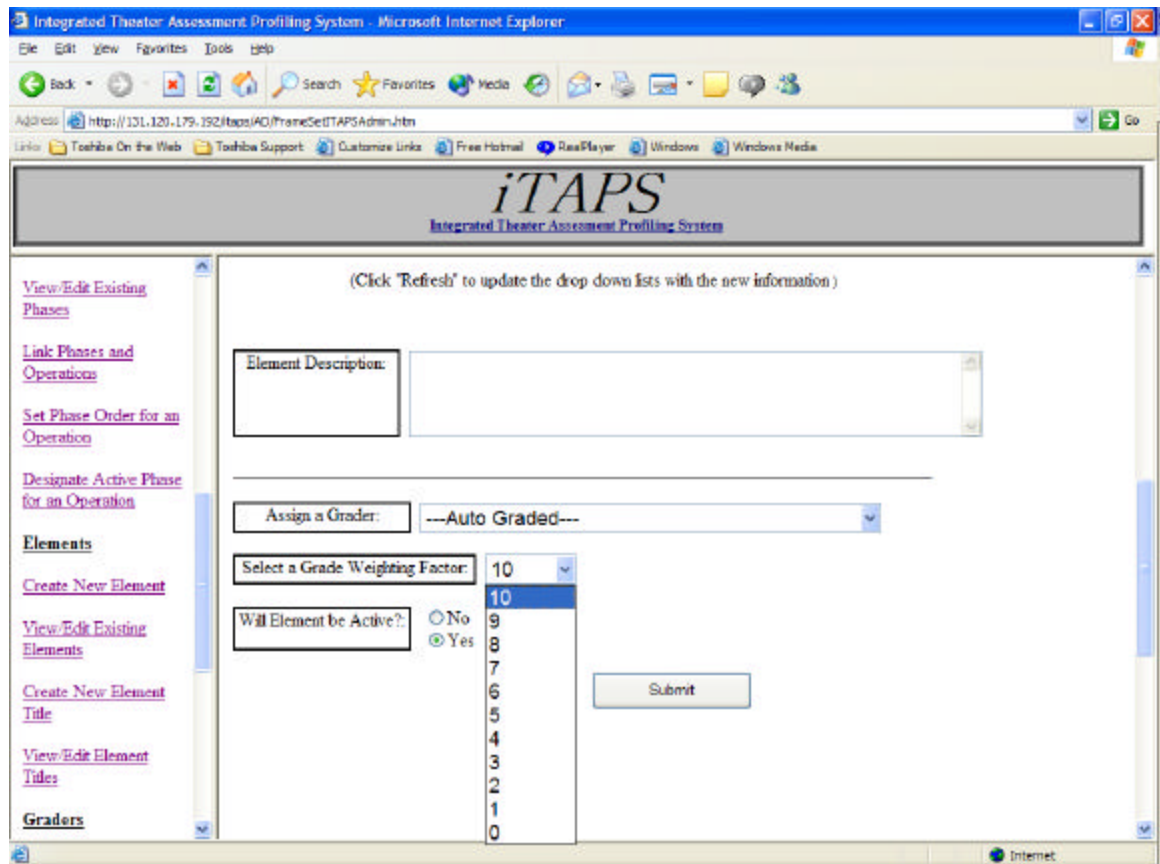


Figure 29. Selecting a Grade Weighting Factor

Each Element has a Grade Weighting Factor that is multiplied by the Grade to give a weighted Grade. This enables the ITAPS user to decide which Elements are more important than others. This gives powerful granularity to the entire ITAPS system. To select a Grade Weighting Factor use the Grade Weighting Factor drop down list. Note: All Grades are normalized to a 0-10 scale upon display.

The “Active” switch is designed to let the user manage which Elements display without deleting them. In this way you can “hide” Elements that senior decision makers decide is not relevant without losing its history or data. You can also create “Standby” Elements that are not yet used but you anticipate might be desired at a later time.

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

Address: http://131.120.179.102/itaps/AD/PhaseSetITAPSAdmin.htm

Links: Toshiba On the Web, Toshiba Support, Customize Links, Free Hotmail, RealPlayer, Windows, Windows Media

iTAPS

Integrated Theater Assessment Profiling System

[Set Phase Order for an Operation](#)

[Designate Active Phase for an Operation](#)

Elements

[Create New Element](#)

[View/Edit Existing Elements](#)

[Create New Element Title](#)

[View/Edit Element Titles](#)

Graders

[Create New Grader](#)

[View/Edit Existing Graders](#)

Create New Element

Instructions: Enter New Element data below. When complete, press "Submit". iTAPS will ask you to confirm your entries. Press "Continue" if the entries are correct, press

Submission Confirmation - Please confirm your data submission:

Element Title: Test Element 1

Element Description: Test Description 1

Linked to Operation: Test Operation Entry

Linked to Phase: Phase Test Entry 1

Parent Element: Phase Test Entry 1 -- Master Element

Side Color: Blue

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

Address: http://131.120.179.102/itaps/AD/PhaseSetITAPSAdmin.htm

Links: Toshiba On the Web, Toshiba Support, Customize Links, Free Hotmail, RealPlayer, Windows, Windows Media

iTAPS

Integrated Theater Assessment Profiling System

[Set Phase Order for an Operation](#)

[Designate Active Phase for an Operation](#)

Elements

[Create New Element](#)

[View/Edit Existing Elements](#)

[Create New Element Title](#)

[View/Edit Element Titles](#)

Graders

[Create New Grader](#)

[View/Edit Existing Graders](#)

Create New Element

Linked to Operation: Test Operation Entry

Linked to Phase: Phase Test Entry 1

Parent Element: Phase Test Entry 1 -- Master Element

Side Color: Blue

Grade Weight: 6

Assigned Grader: ---Auto Graded---

Will Phase be Active?: Yes

Continue Cancel

Figure 30. Confirming your entry

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 30). If you made a mistake or do not wish to create this Phase entry, Click "Cancel". You will be returned to the previous screen. If the entry is correct, click "Continue" and ITAPS will update its database.

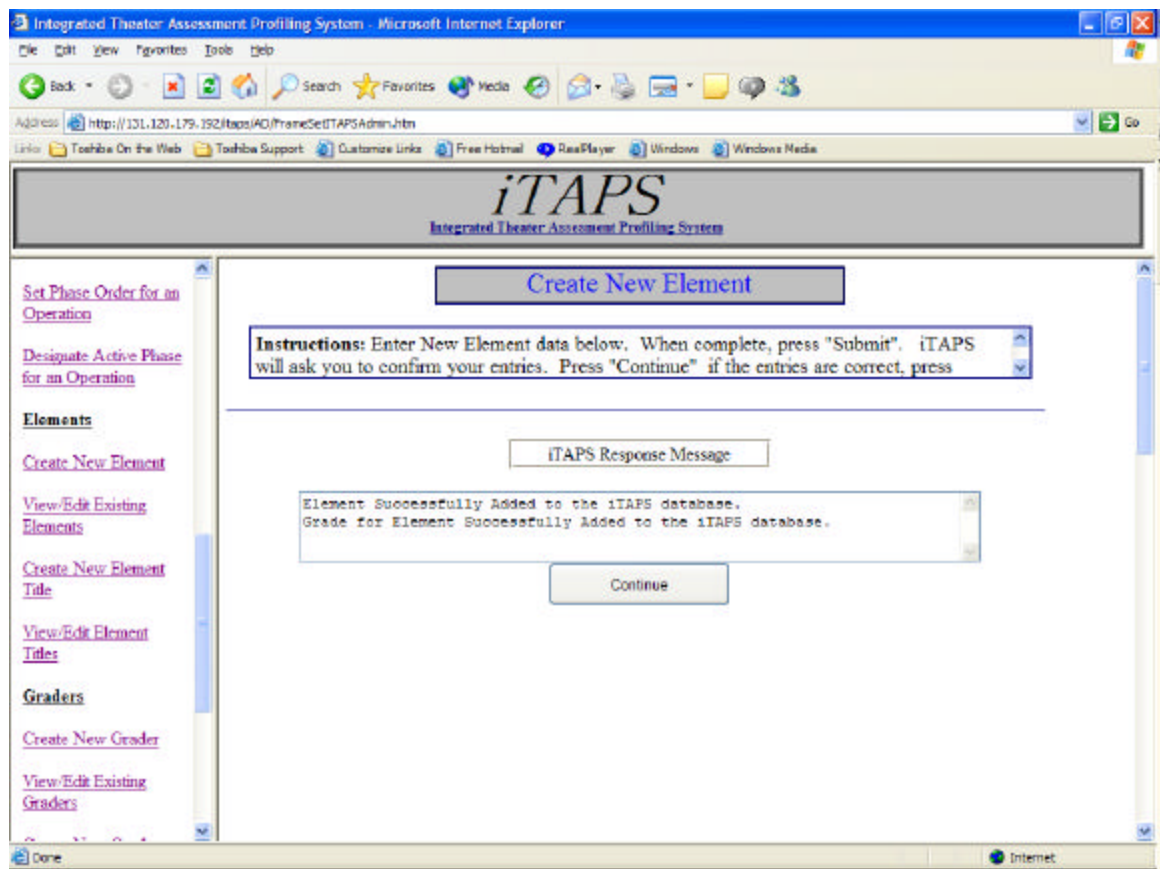


Figure 31. ITAPS Responds

ITAPS will display a response screen concerning your submission (see figure 31). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.6 Editing/Deleting/Linking Elements

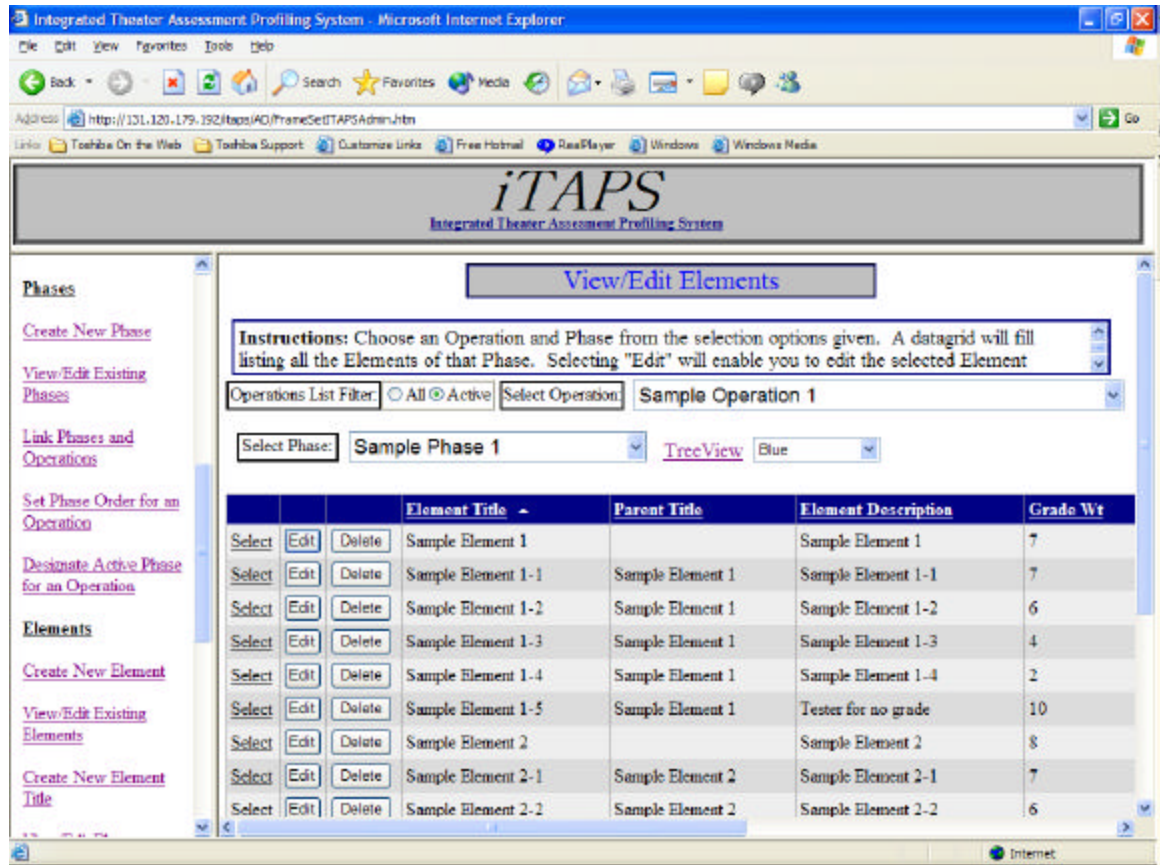


Figure 32. Viewing and Editing Elements

To view, edit or delete Elements already in ITAPS select "View/Edit Elements" from the Admin Tools menu (see figure 32). Select the Operation and Phase you wish to view Elements for by using the Operation and Phase drop down lists and the Operation List Filter. The Element information is displayed in a tabular grid format. The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Element click the "Create New Element" hyperlink. When you are done creating the new Element click "Refresh" to update the datagrid with your new entry. Once a valid Operation and Phase are entered a "TreeView" hyperlink becomes available that allows you to view a graphical tree of the current Element structure in a separate window. Select the sidecolor of the Elements you wish to view and click on the "TreeView" hyperlink.

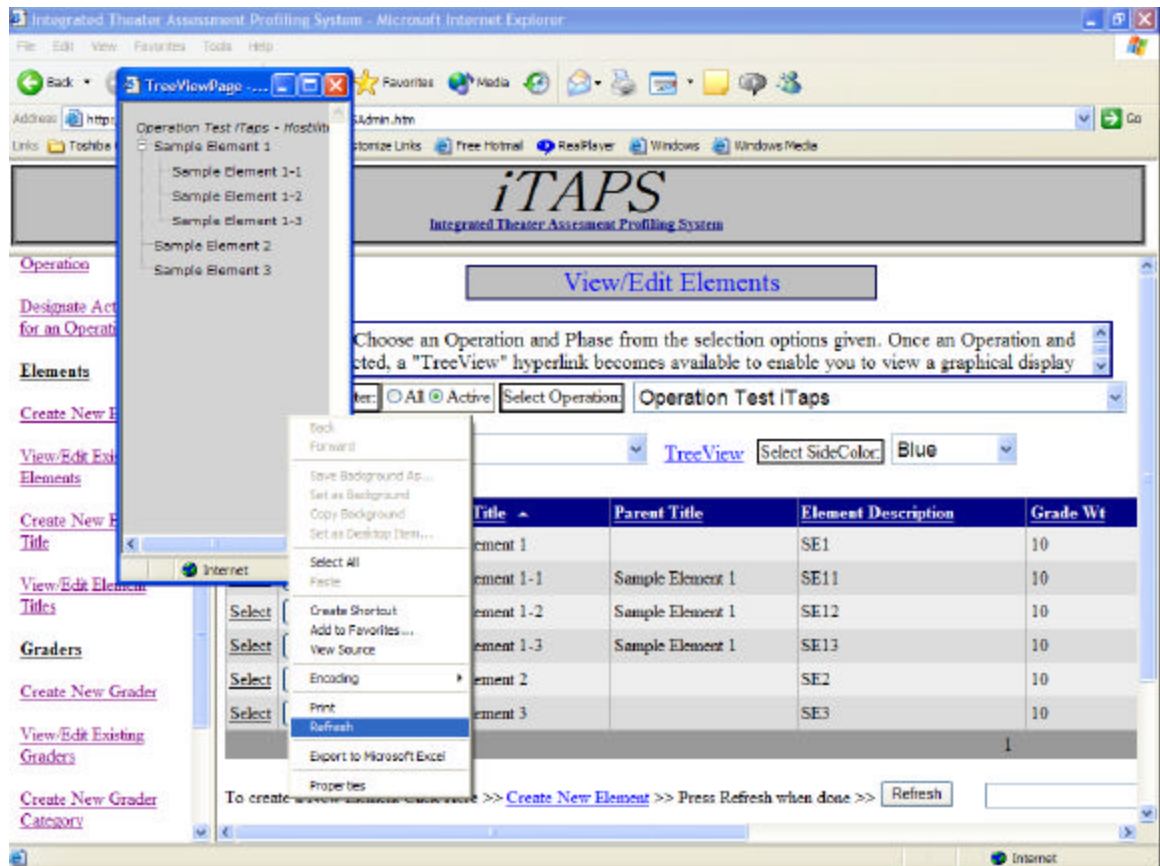


Figure 32a. Viewing and Editing Elements

You can resize and position this treeview window wherever it is convenient. Remember to refresh the treeview after every change in Operation Element structure (see Figure 32b). To refresh the treeview, right click inside the view and select “Refresh”. (You can also click on the Elements in the treeview and view the radar diagrams from here as well.)

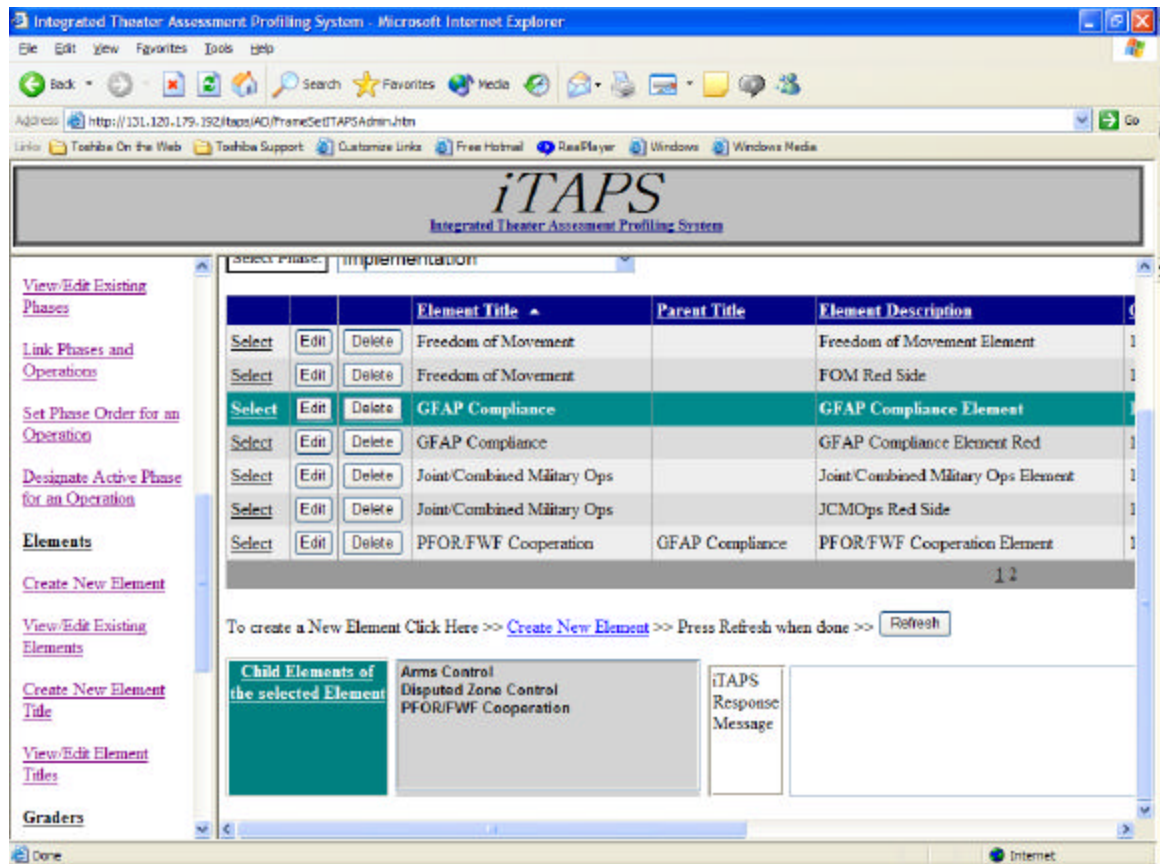


Figure 33. Selecting an Element

Child Elements of the Selected Element will be listed in the listbox below the datagrid when a particular Element is selected. Elements that have Child Elements cannot be deleted.

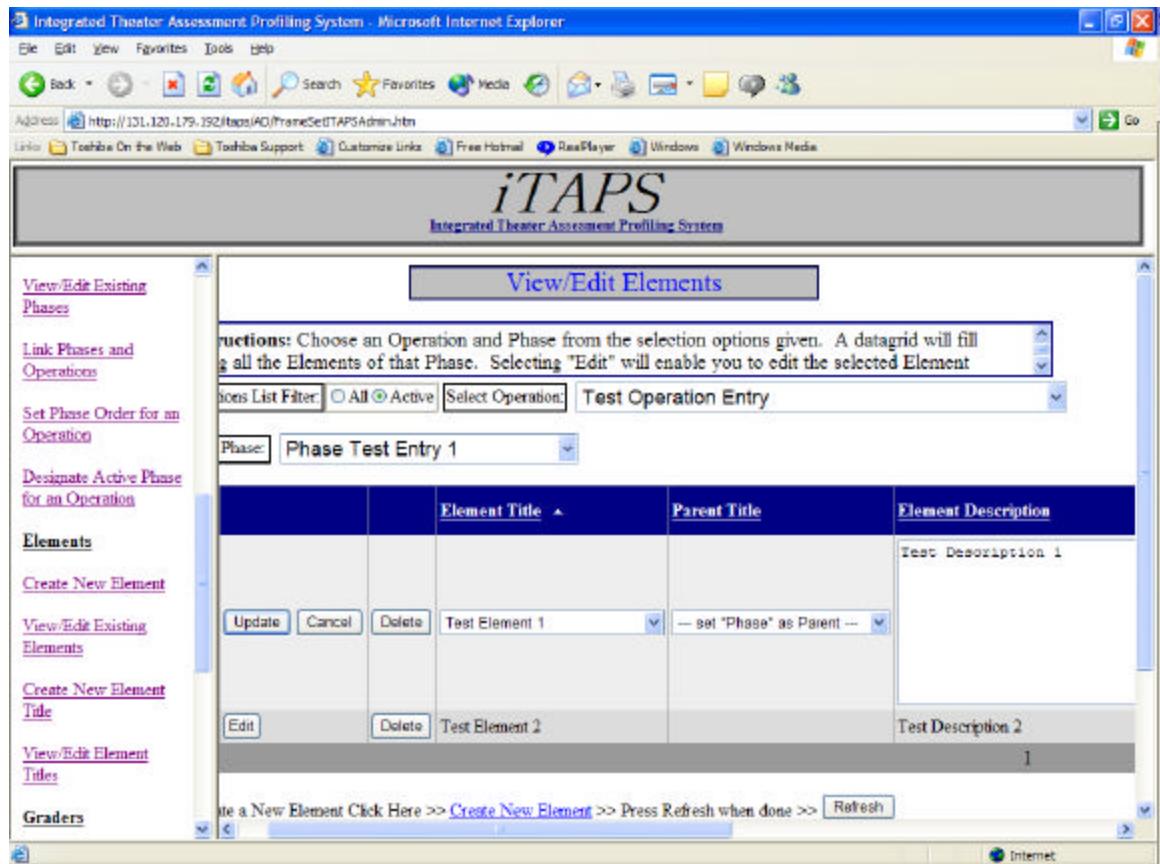


Figure 34. Editing Elements

To Edit an Element, click the "Edit" button for that Element. An Edit Template will be displayed for that Element (see figure 34). Make the changes you desire and click "Update" to enter your changes or "Cancel" to cancel your changes.

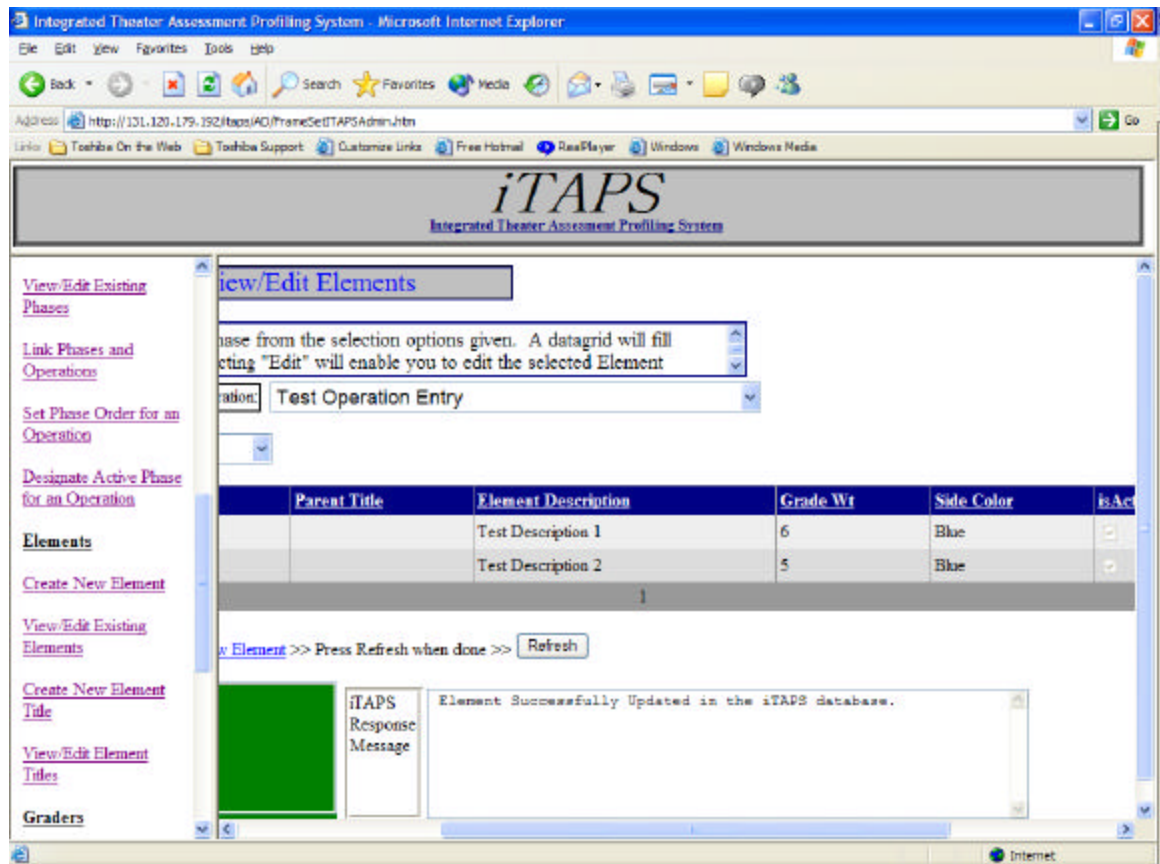


Figure 35. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 35). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

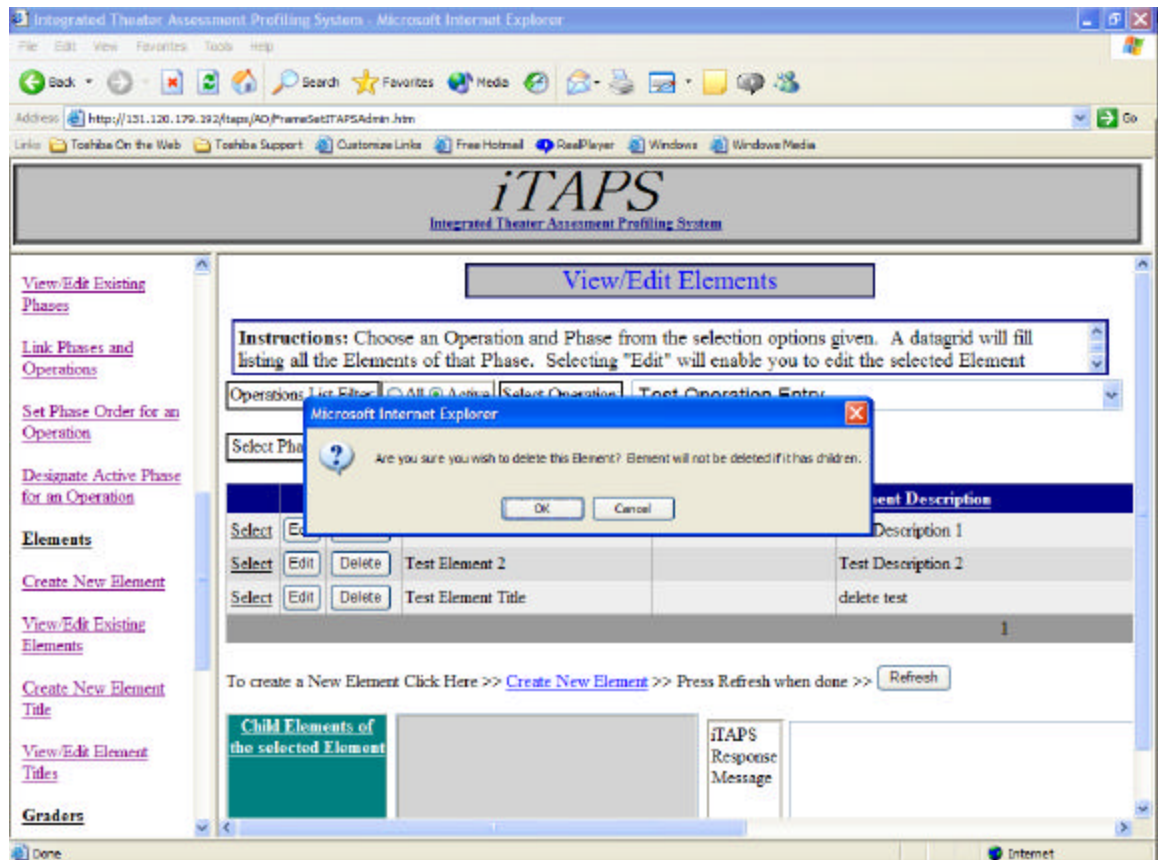


Figure 36. Deleting an Element

To Delete an Element, click the "Delete" button for the desired Element. A confirmation box will ask you to confirm your selection (see figure 36). Clicking "OK" will cause the Element to be permanently deleted if it has no children. Clicking "Cancel" will cancel your delete request. Elements that have Child Elements cannot be deleted.

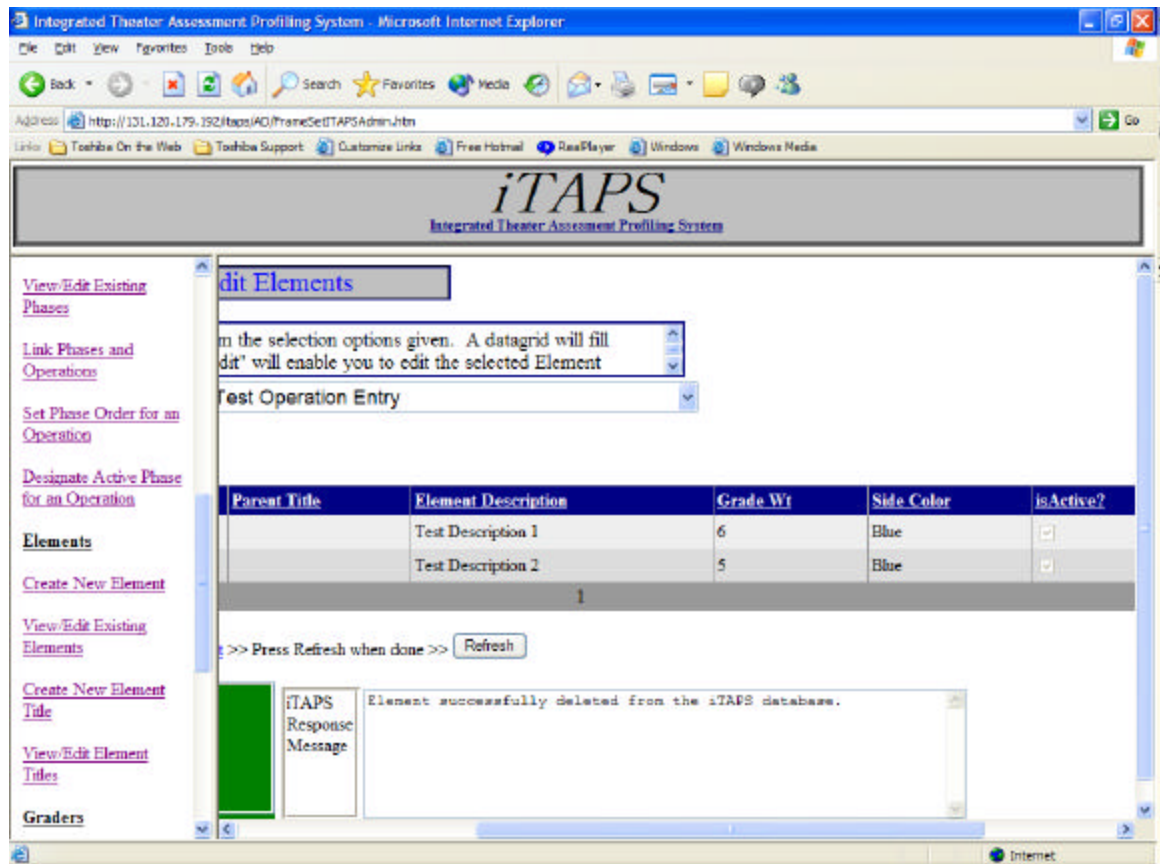


Figure 37. ITAPS Responds

ITAPS will display a response screen concerning your submission (see figure 37). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.7 Linking Operations/Phases

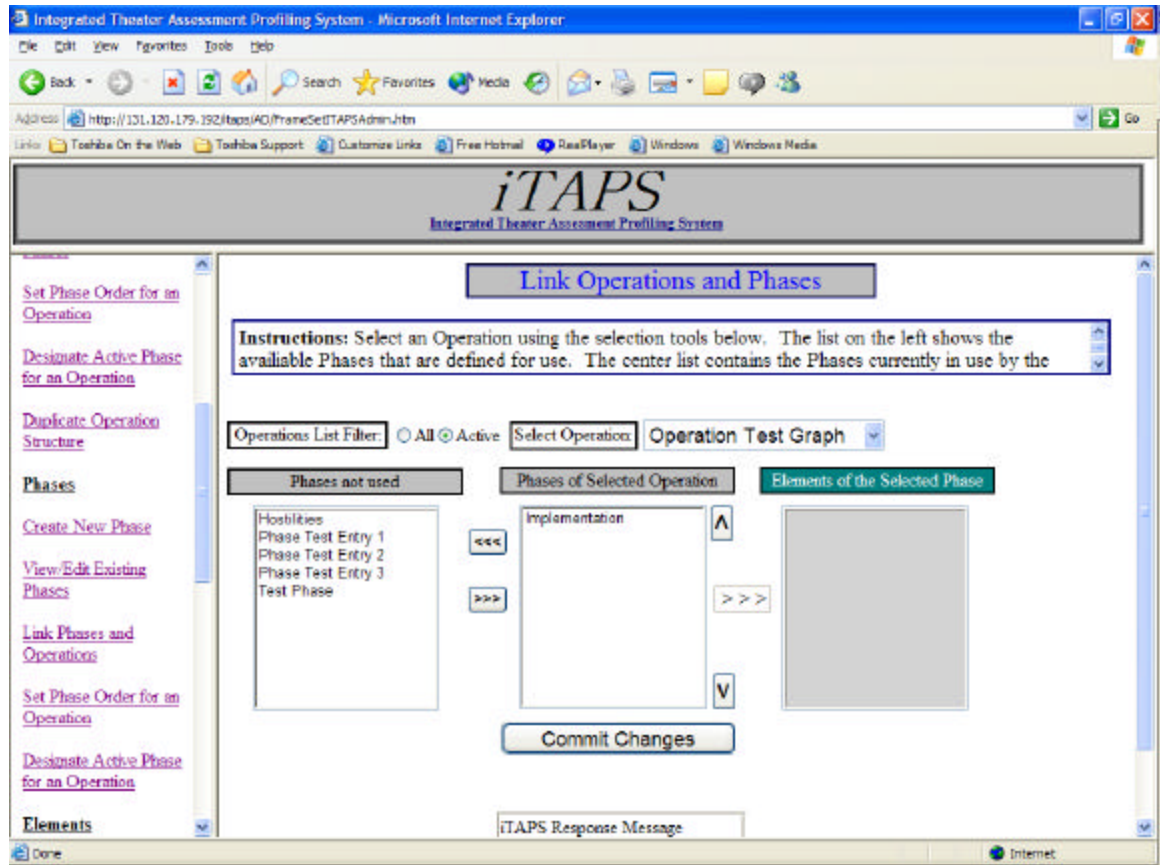


Figure 38. Linking Operations and Phases

To Link Operations and Phases, select “Link Operations and Phases” or “Link Phases and Operations” from the Admin Tools menu (see figure 38). Select the Operation you wish to select Phases for using the Operation List Filter and the Operation drop down list.

There are three lists on this page. The leftmost list is a list of Phases the current Operation is *not* using. The center list is a list of Phases the current Operation is using. Using the arrow keys Phases may be moved between the two lists and may be reordered in the center list via the up and down arrows. The list to the right lists the Elements that are within the currently selected Phase. You cannot remove a Phase from an Operation if it has Elements.

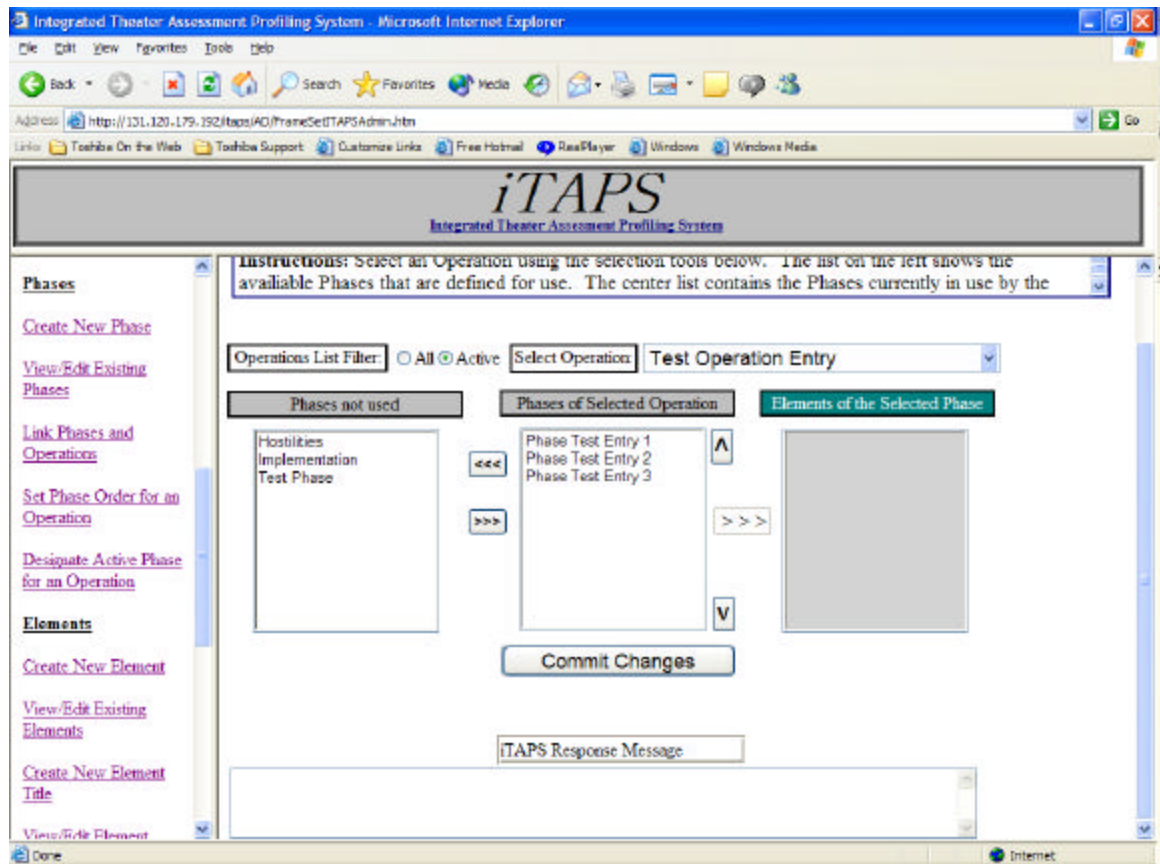


Figure 39. Linking Operations and Phases

Once you have placed all the Phases you desire into the center list, press “Commit Changes” to process your request.

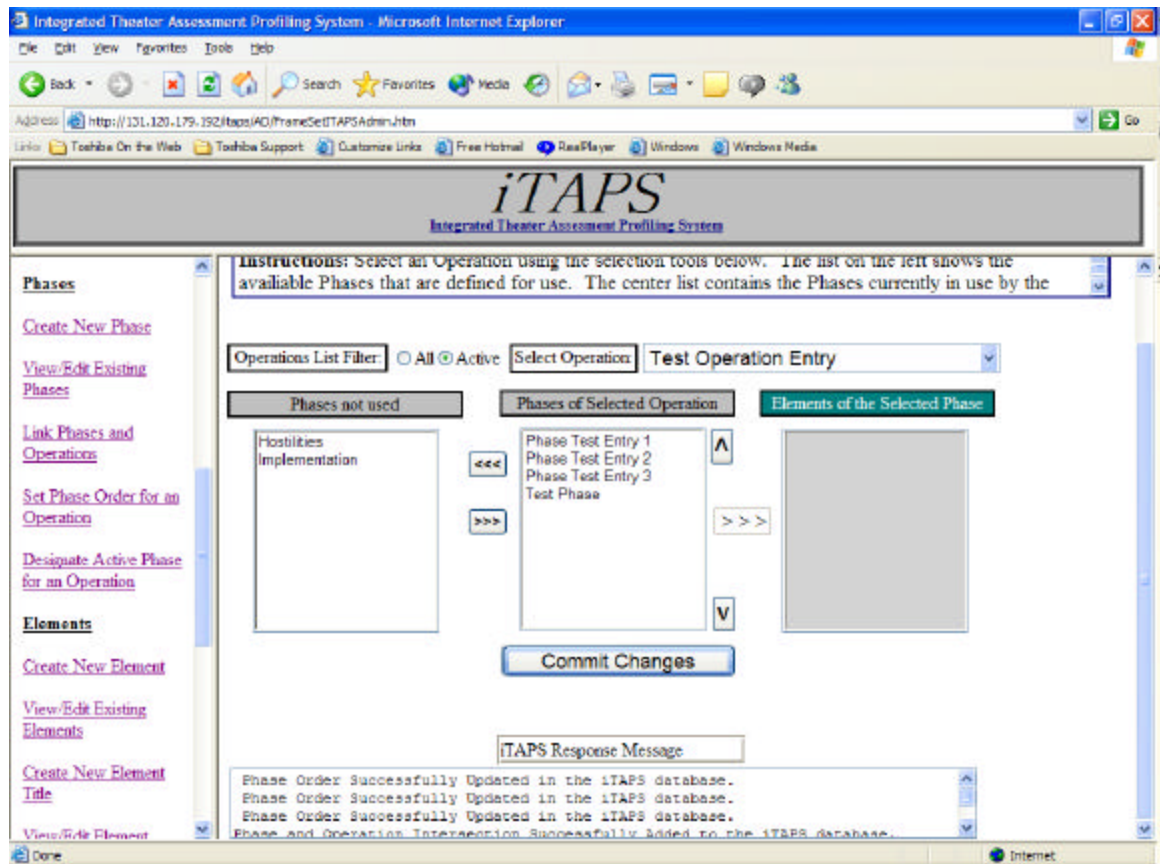


Figure 40. ITAPS Responds

ITAPS will display a response screen concerning your submission (see figure 40). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.8 Designating Active Phases

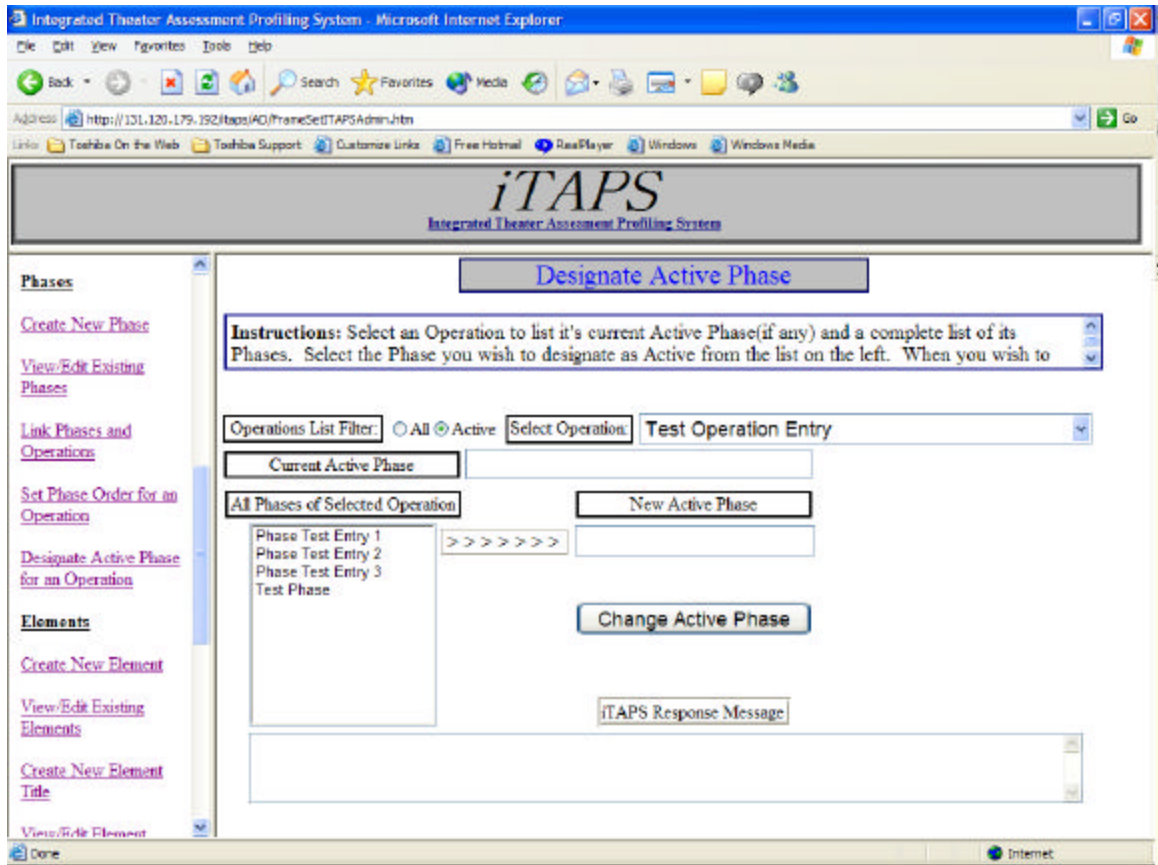


Figure 41. Designating the Active Phase

To designate which Phase of a particular Operation is the Active Phase, select "Designate Active Phase for an Operation" from the Admin Tools menu (see figure 41). Select the Operation you wish to designate the Active Phase for using the Operation List Filter and the Operation drop down list.

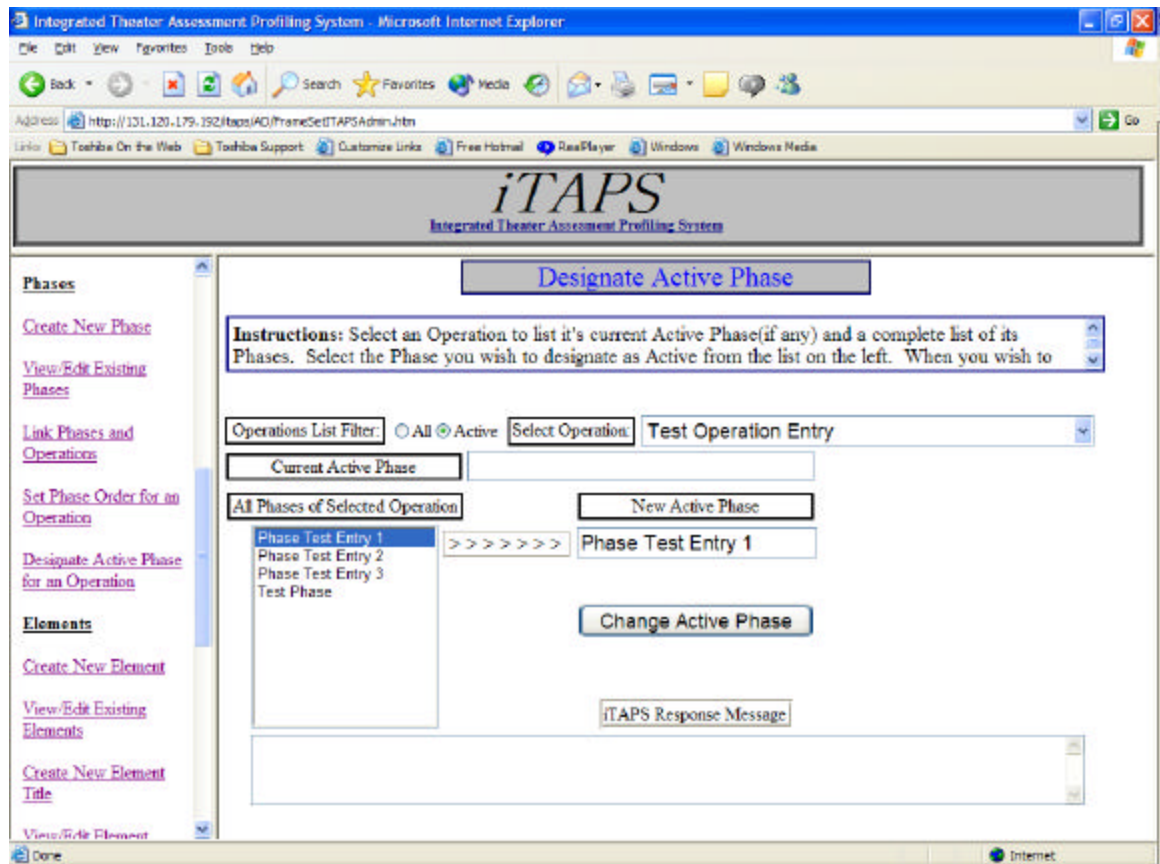


Figure 42. Designating the Active Phase

To select the Active Phase, click on the desired Phase from the left list box. ITAPS will update the “New Active Phase” text box with your selection. Click the “Change Active Phase” button to update the Active Phase.

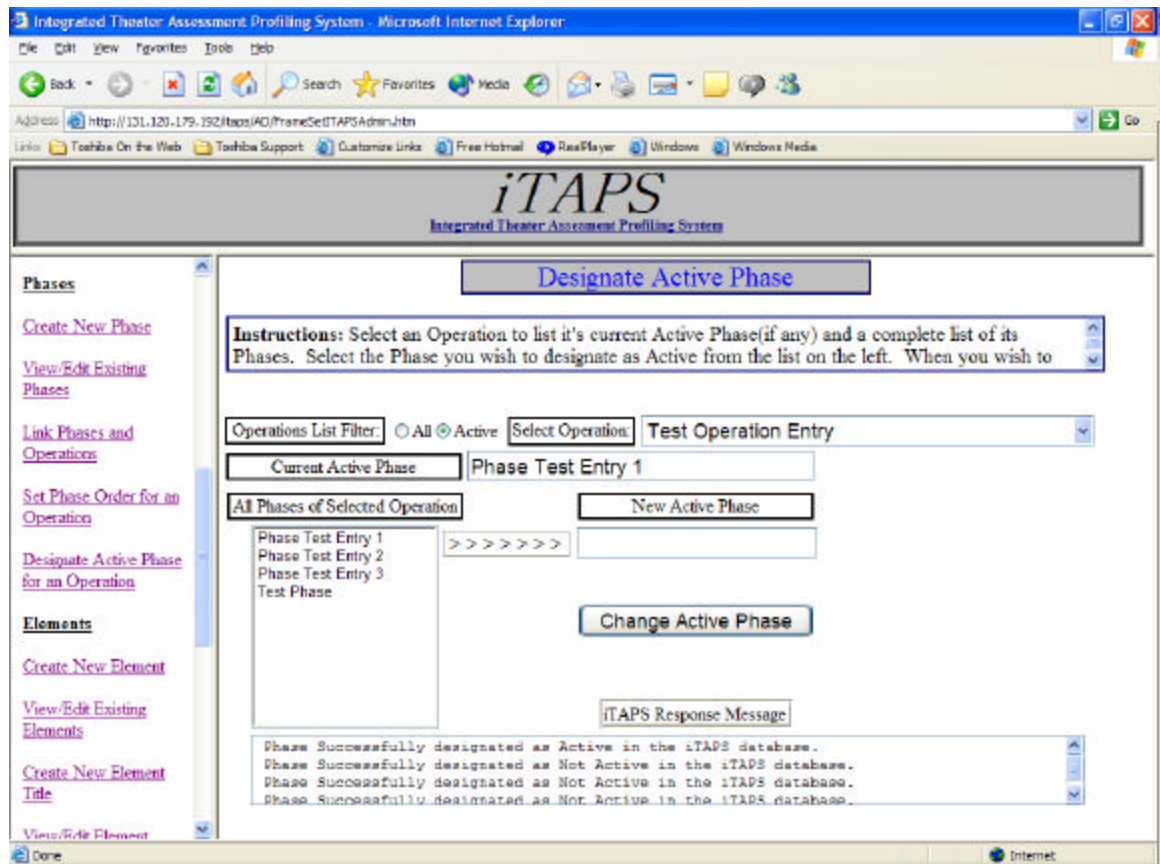


Figure 43. ITAPS Responds

ITAPS will update the Current Active Phase text box with your newly designated Active Phase (see figure 43). ITAPS will also display a response message concerning your submission (see figure 43). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.9 Ordering Phases of an Operation

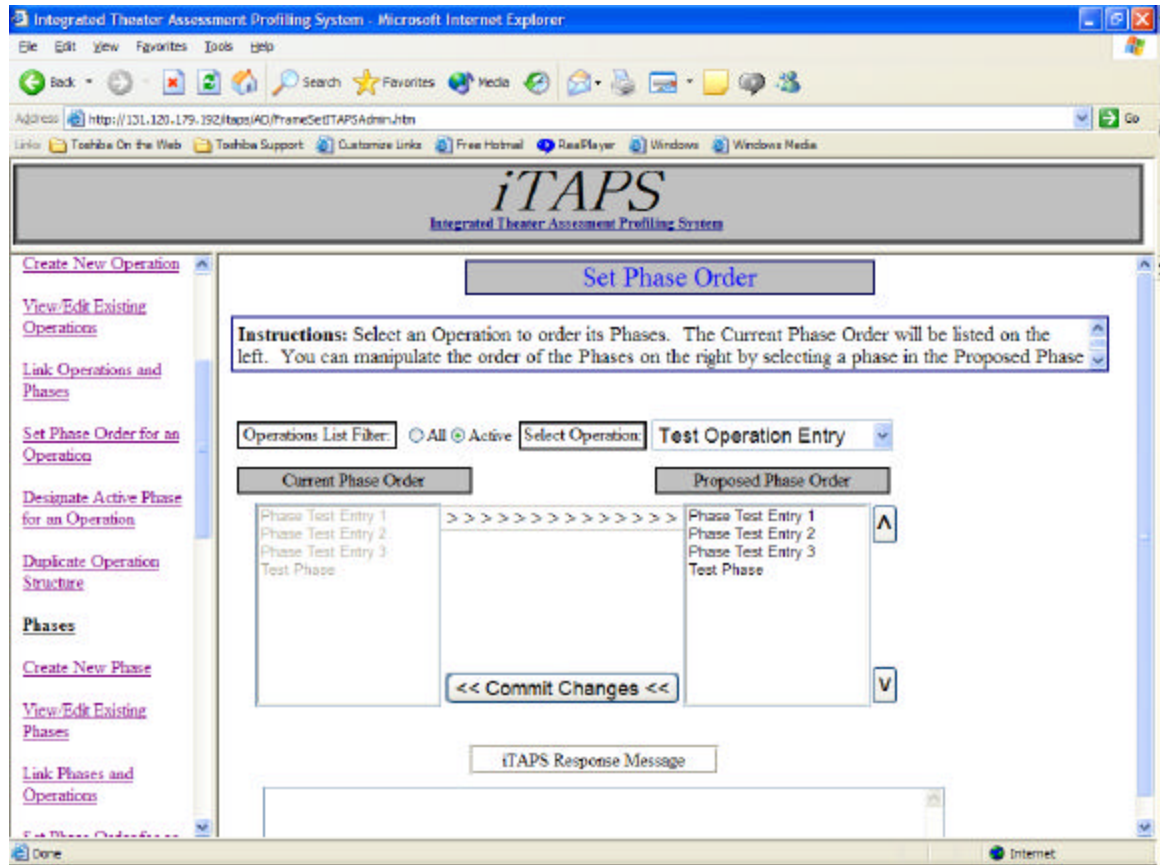


Figure 44. Setting the Phase Order

To set the Phase Order of a particular Operation, select “Set Phase Order for Operation” from the Admin Tools menu (see figure 44). Select the Operation you wish to set the Phase Order for by using the Operation List Filter and the Operation drop down list. The current phase order for the selected Operation will fill both list boxes. The list on the right is editable.

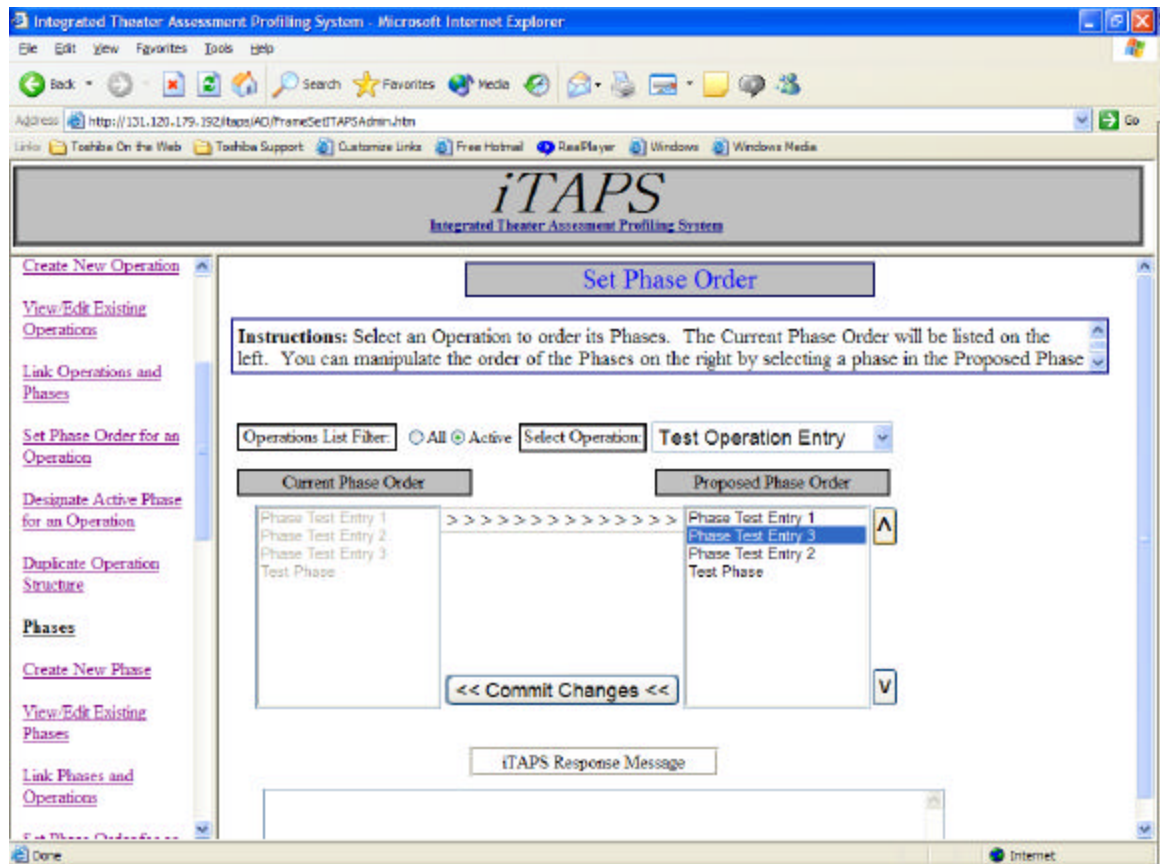


Figure 45. Setting the Phase Order

To reorder the Phases, select a Phase in the list on the right and use the up/down arrows to change its position. The left list box indicates the actual Phase Order. When you are satisfied with your Proposed Phase Order, click the "Commit Changes" button to make your changes permanent.

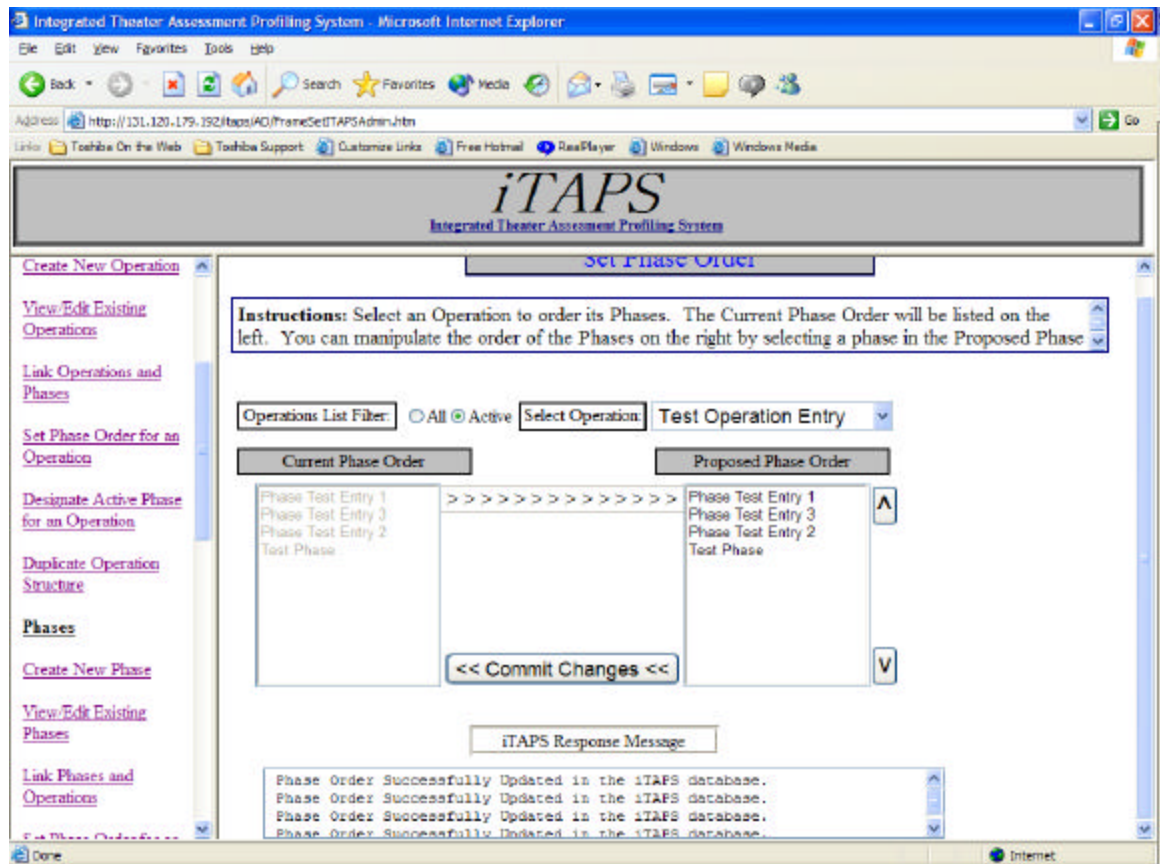


Figure 46. ITAPS Responds

ITAPS will update the Phase Order with your Proposed Phase Order (see figure 46). ITAPS will also display a response message concerning your submission (see figure 46). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.10 Creating Element Titles

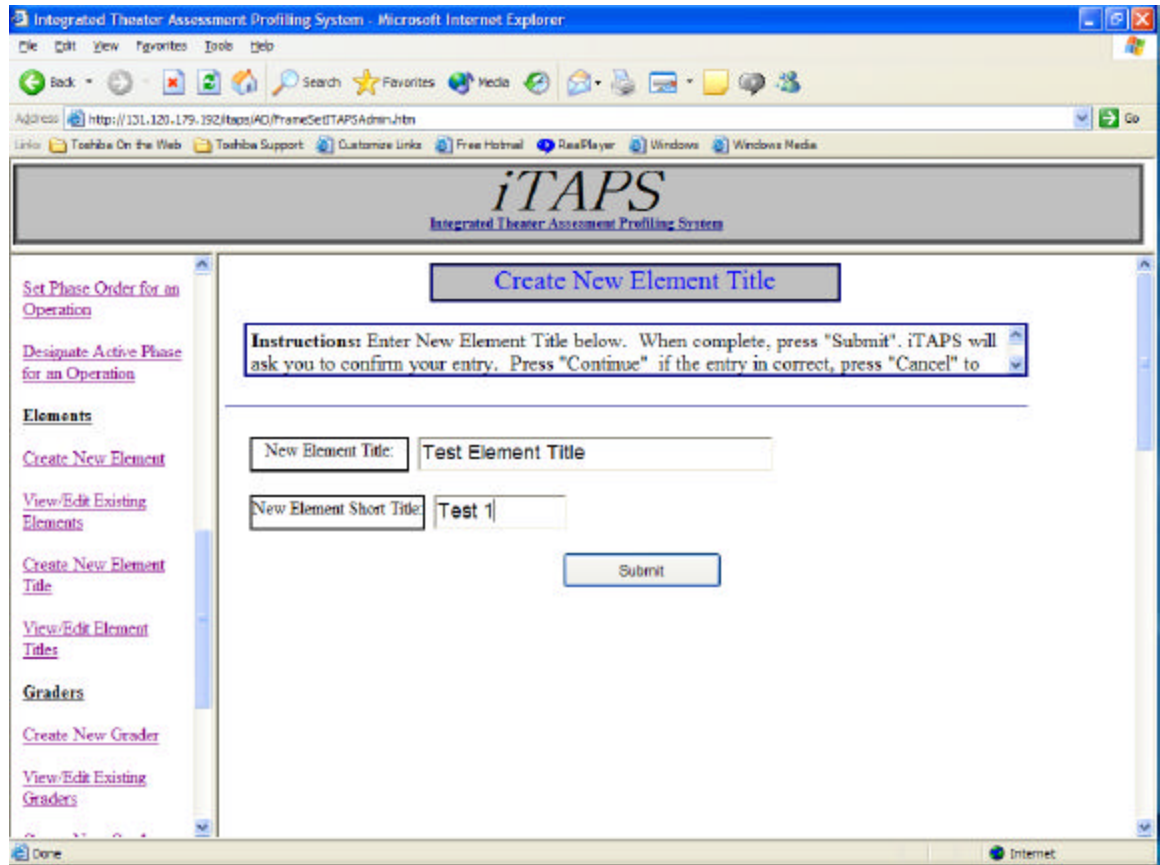


Figure 47. Creating an Element Title

To create an Element Title, select “Create New Element Title” from the Admin Tools menu (see figure 47). Both entries are mandatory. Click the “Submit” button when satisfied with your entry.

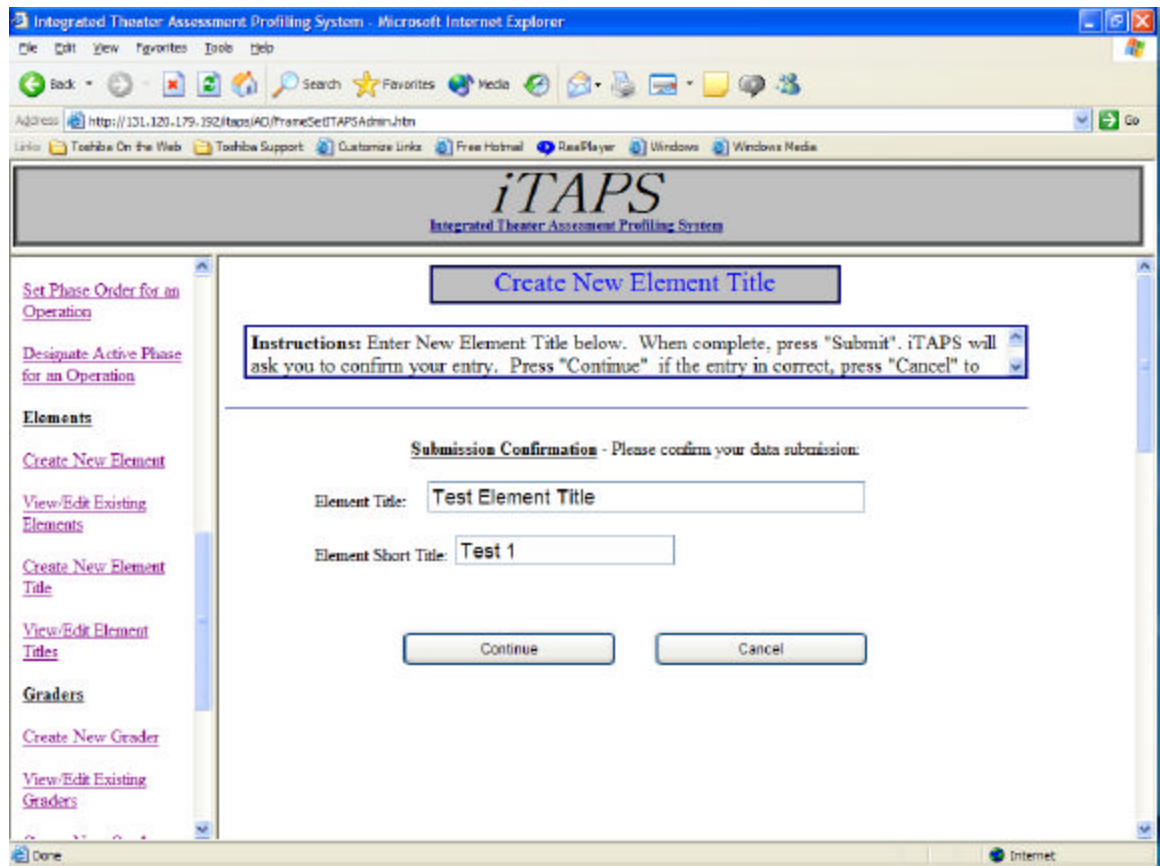


Figure 48. Confirming your entry

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 48). If you made a mistake or do not wish to create this Element Title entry, Click "Cancel". You will be returned to the previous screen. If the entry is correct, click "Continue" and ITAPS will update its database.

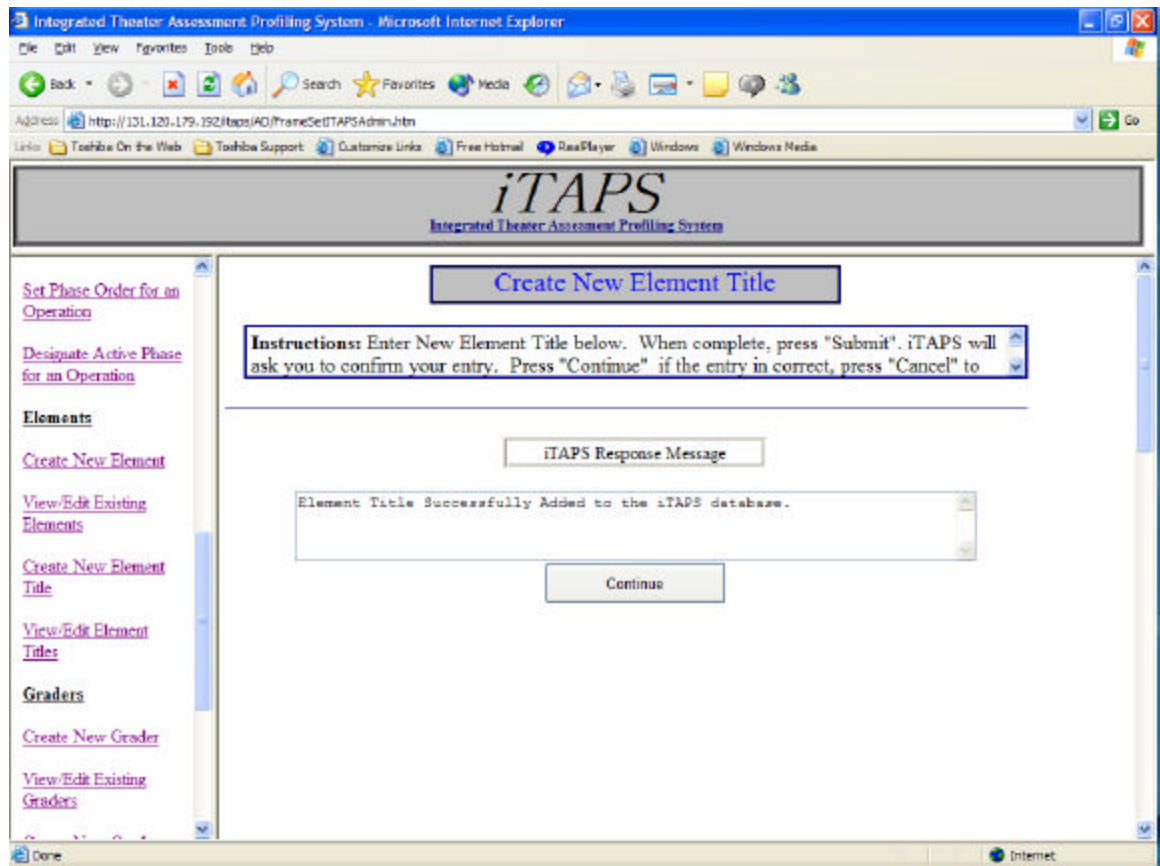


Figure 49. ITAPS Responds

ITAPS will display a response message concerning your submission (see figure 49). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.11 Editing/Deleting Element Titles

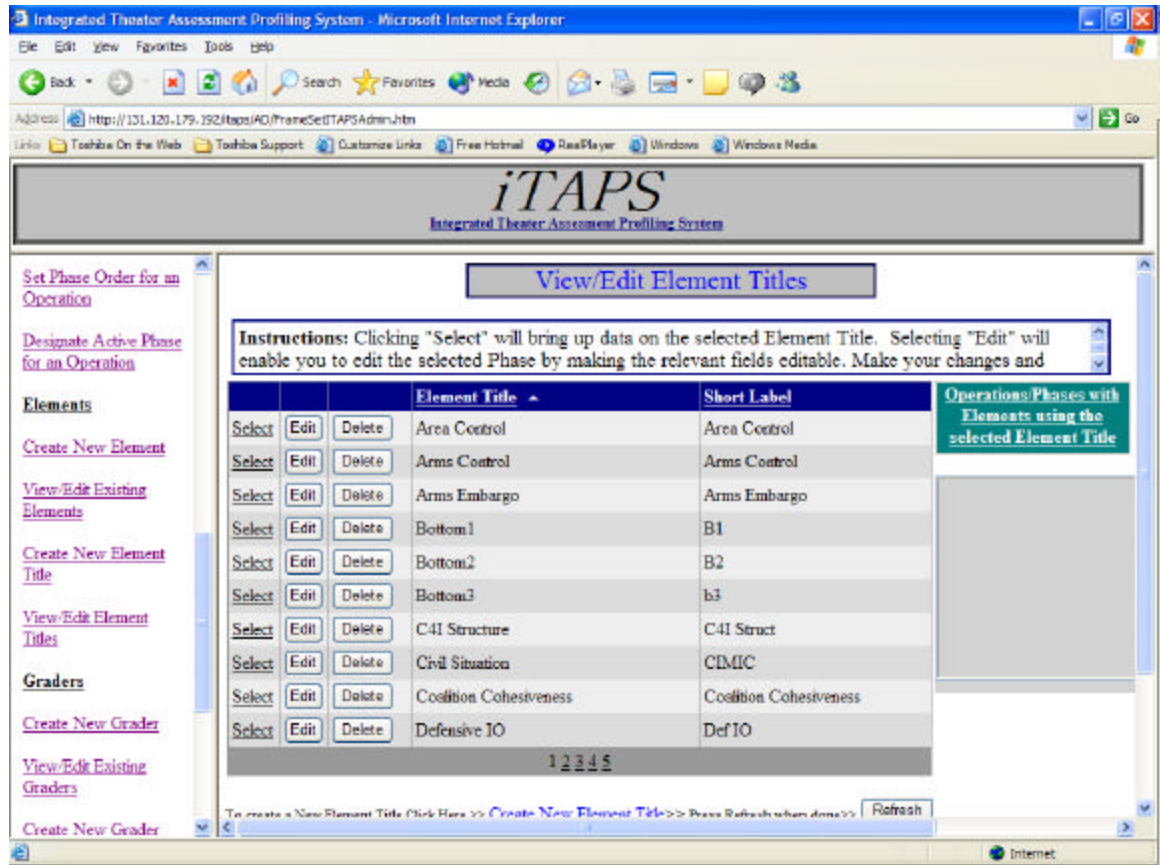


Figure 50. Viewing and Editing Element Titles

To view, edit or delete Element Titles already in ITAPS select "View/Edit Element Titles" from the Admin Tools menu (see figure 50). The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Element Title, click the "Create New Element Title" hyperlink. When you are done creating the new Element Title, click "Refresh" to update the datagrid with your entry.

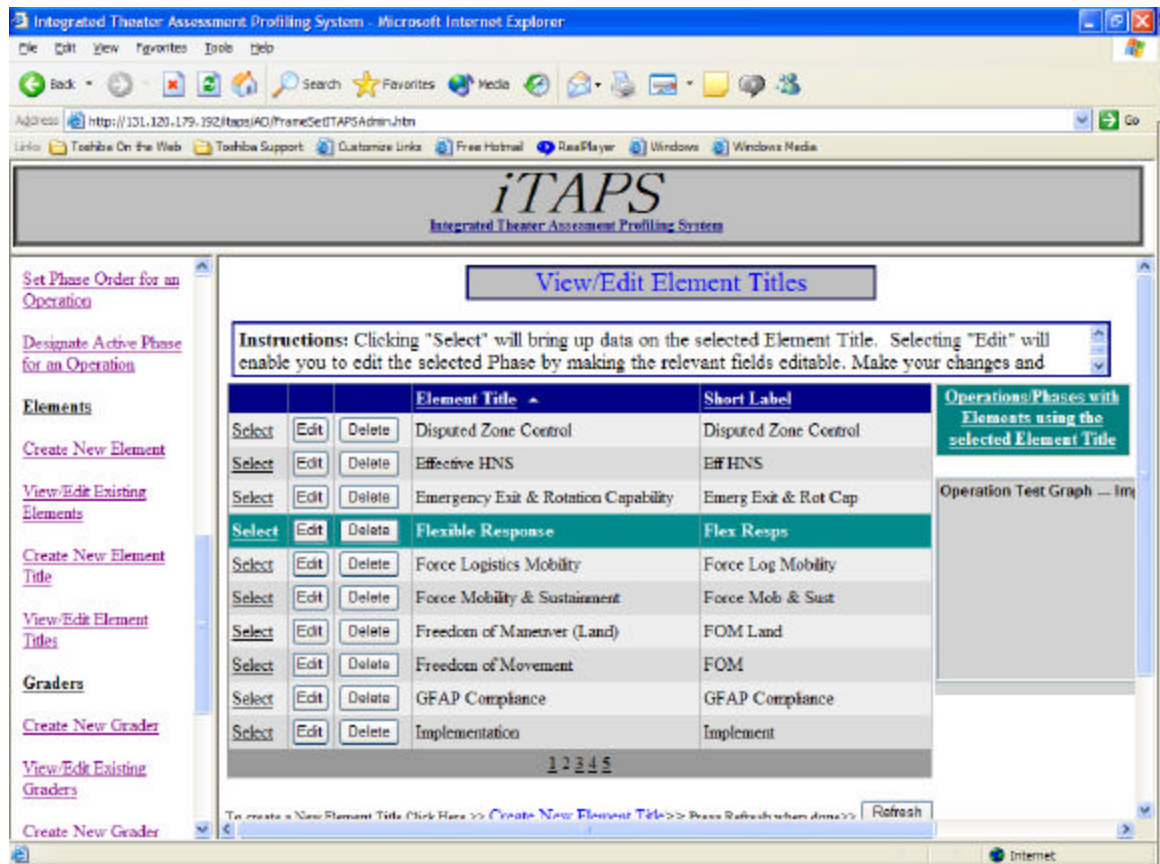


Figure 51. Selecting an Element Title

Operations/Phases that use the selected Element Title will be displayed in the list box to the left (see figure 51). Scroll the screen to the right to see the complete Operation/Phase names. Element Titles in use by an Operation/Phase cannot be deleted.

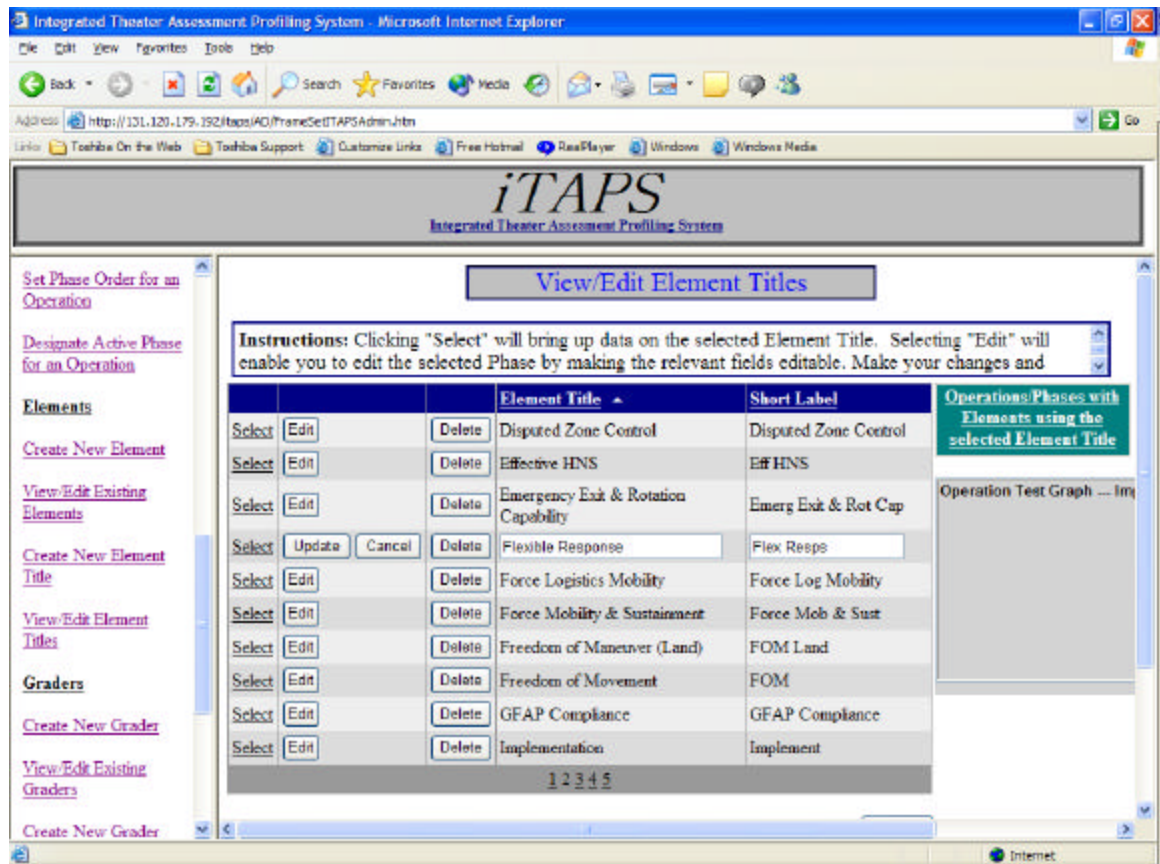
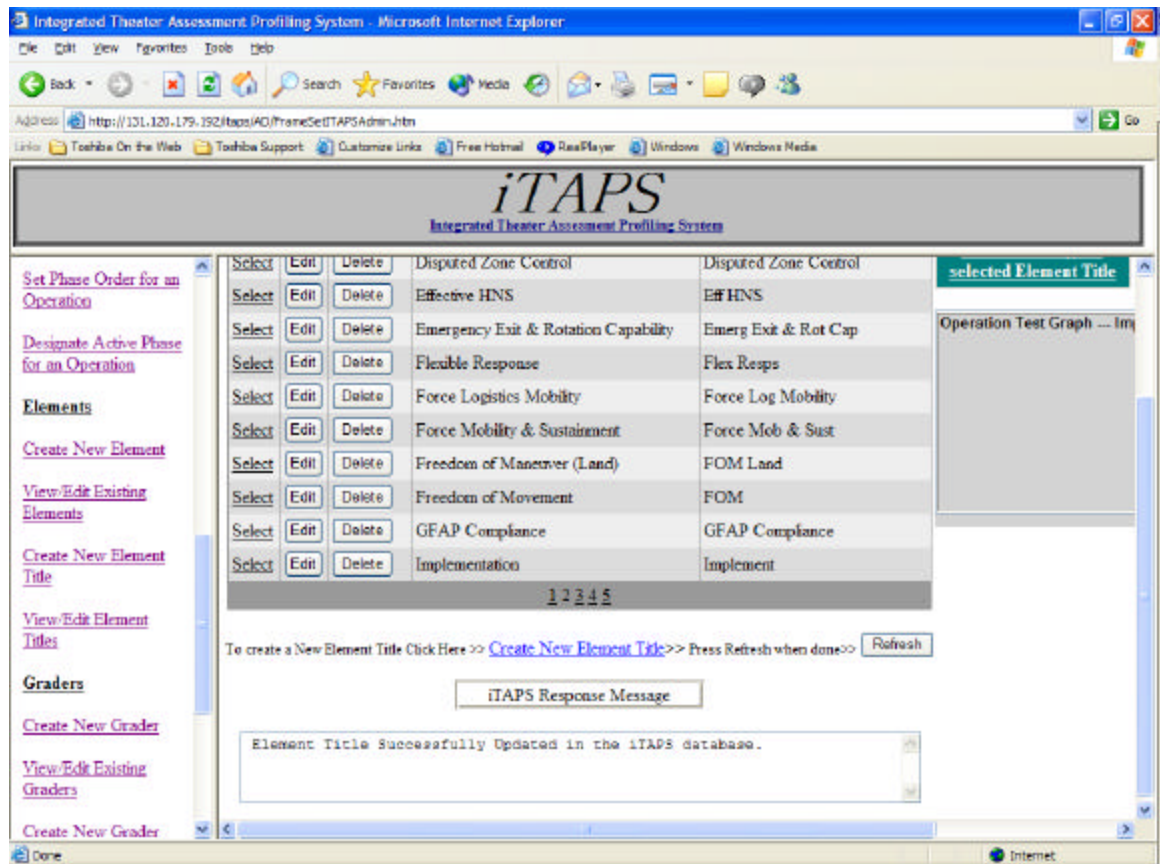


Figure 52. Editing an Element Title

To Edit an Element Title, click the “Edit” button for that Element Title. An Edit Template will be displayed for that Element Title (see figure 52). Make the changes you desire and click “Update” to enter your changes or “Cancel” to cancel your changes.



. Figure 53. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 53). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

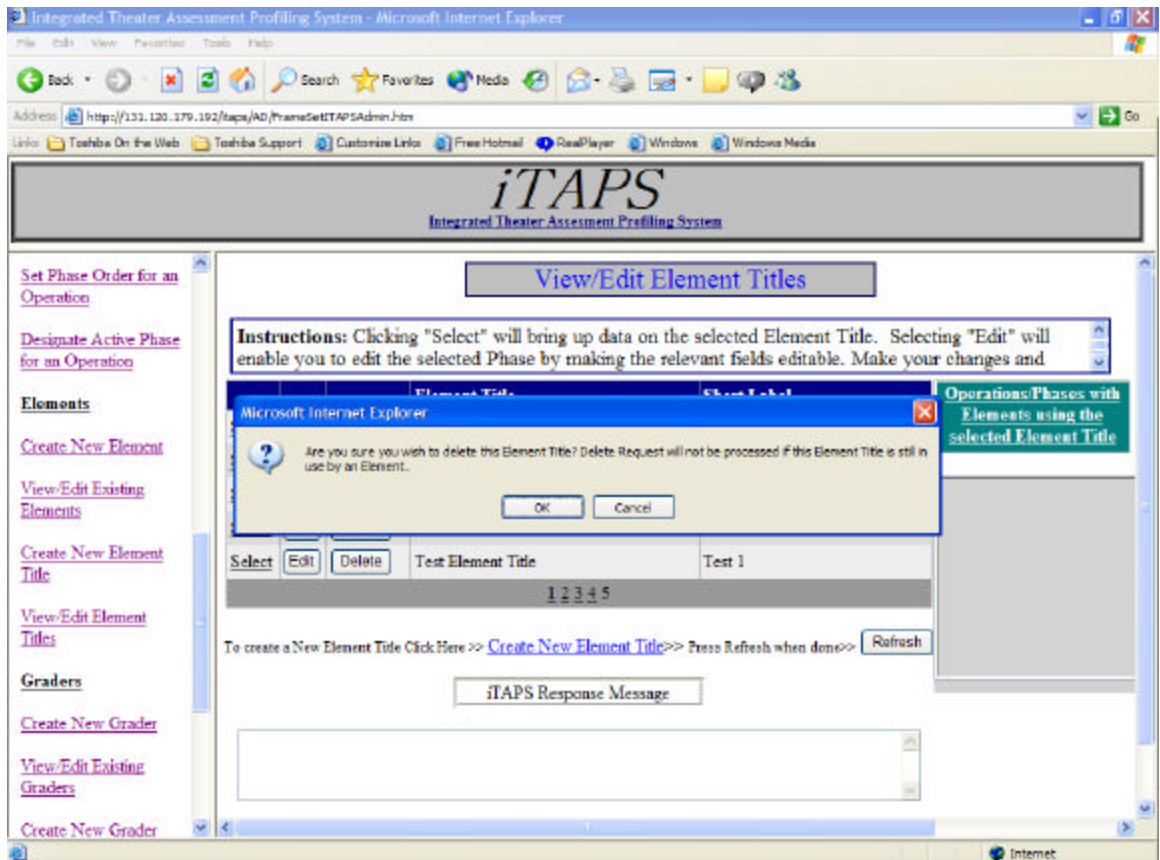
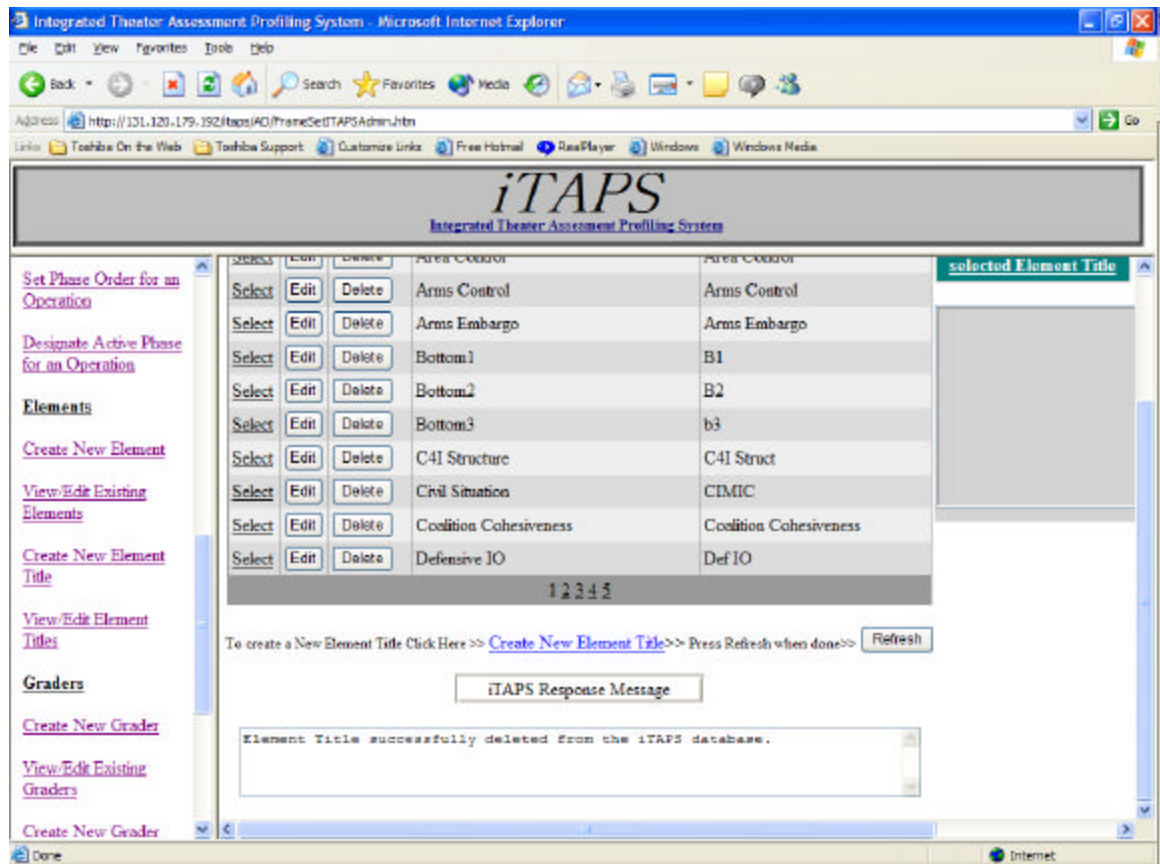


Figure 54. Deleting an Element Title

To Delete an Element Title, click the “Delete” button for the desired Element Title. A confirmation box will ask you to confirm your selection (see figure 54). Clicking “OK” will cause the Element to be permanently deleted if it has no children. Clicking “Cancel” will cancel your delete request. Element Titles in use by an Operation/Phase cannot be deleted.



. Figure 55. ITAPS Responds

ITAPS will display a response message concerning your submission (see figure 55). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.12 Creating Graders

The screenshot shows a web browser window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows the URL "http://131.129.179.192/itaps/AD/FrameSetITAPSAdmin.htm". The page features a header with the "iTAPS" logo and the text "Integrated Theater Assessment Profiling System". A left sidebar contains a menu with links: "Set Phase Order for an Operation", "Designate Active Phase for an Operation", "Elements", "Create New Element", "View/Edit Existing Elements", "Create New Element Title", "View/Edit Element Titles", "Graders", "Create New Grader", and "View/Edit Existing Graders". The main content area is titled "Create New Grader Entry" and contains an instruction box: "Instructions: Enter New Grader Data below. When complete, press 'Submit'. iTAPS will ask you to confirm your entry. Press 'Continue' if the entry is correct, press 'Cancel' to". Below this is a section titled "Grader Information" with a dropdown menu labeled "Select a Grader Category:" and a "Refresh" button. A note below the dropdown says: "<< Select from the drop down list above OR Create New >> [Create a New Grader Category](#) (Press Refresh after creating a new Grader Category to update the drop down list)". At the bottom, there are three text input fields labeled "Grader User ID:", "First Name:", and "Last Name:".

Figure 56. Creating a Grader

To create a new Grader, select "Create New Grader" from the Admin Tools menu (see figure 56). The Grader Category and Grader User ID entries are mandatory. All other entries are optional.

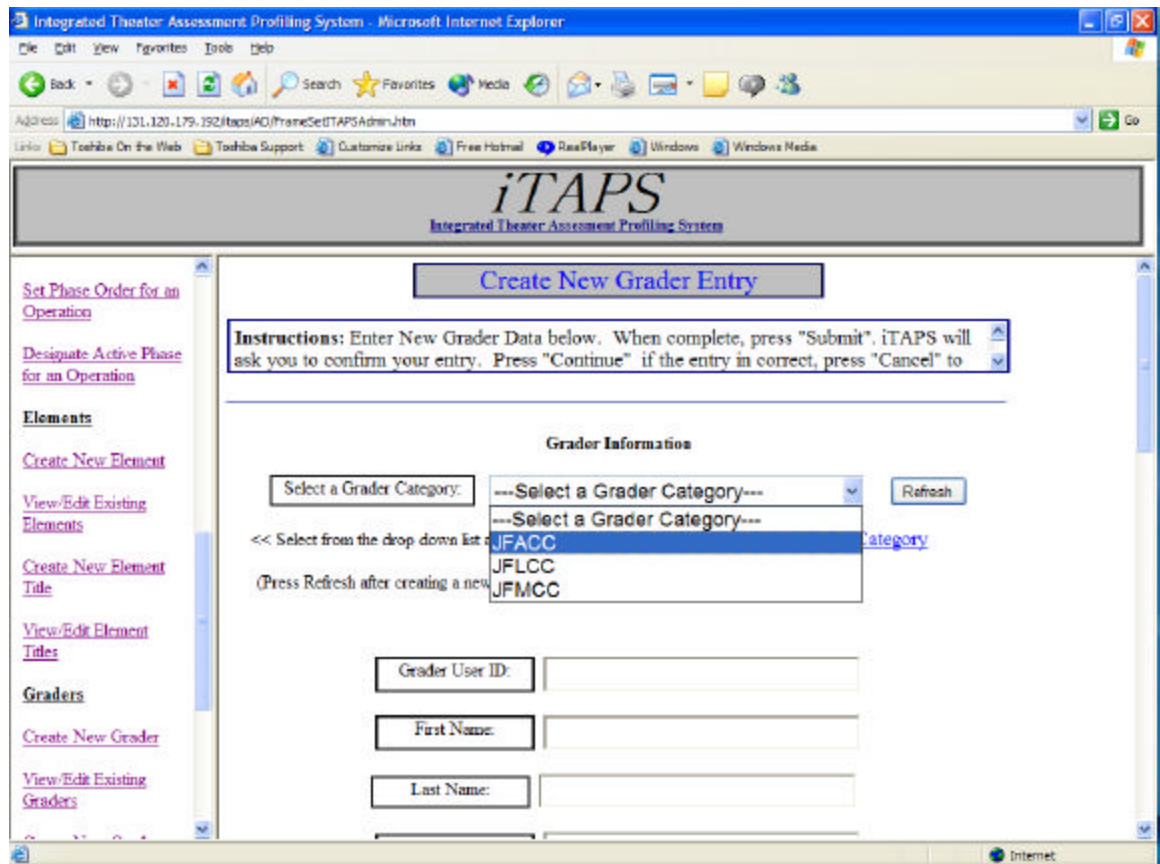


Figure 57. Creating a Grader

Use the Grader Category drop down list to select the Grader Category this Grader will belong to. If you need a new Grader Category for this Grader, click on the “Create a New Grader Category” hyperlink. Click “refresh” when done creating the new Grader Category in order to update the drop down list with the new entry.

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

Address: <http://131.120.179.192/itaps/AD/FrameSetITAPSAdmin.htm>

iTAPS
Integrated Theater Assessment Profiling System

Create New Grader Entry

Instructions: Enter New Grader Data below. When complete, press "Submit". iTAPS will ask you to confirm your entry. Press "Continue" if the entry is correct, press "Cancel" to

Grader Information

Select a Grader Category:

<< Select from the drop down list above OR Create New >> [Create a New Grader Category](#)

(Press Refresh after creating a new Grader Category to update the drop down list)

Grader User ID:

First Name:

Last Name:

Rank:

Command:

Department:

Office Phone:

Commercial Phone:

Cell Phone:

Pager:

Email Address:

Figure 58. Creating a Grader

Click the "Submit" button when satisfied with your entry.

The image consists of two screenshots of the iTAPS web application interface, showing the process of creating a new grader entry.

Top Screenshot: Create New Grader Entry

The browser window title is "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.120.179.192/ITAPSAdmin/". The page header displays the iTAPS logo and "Integrated Theater Assessment Profiling System".

On the left sidebar, there are links for "Set Phase Order for an Operation", "Designate Active Phase for an Operation", "Elements", "Create New Element", "View/Edit Existing Elements", "Create New Element Title", "View/Edit Element Titles", "Graders", "Create New Grader", and "View/Edit Existing Graders".

The main content area has a "Create New Grader Entry" button. Below it, an instruction box states: "Instructions: Enter New Grader Data below. When complete, press 'Submit'. iTAPS will ask you to confirm your entry. Press 'Continue' if the entry is correct, press 'Cancel' to".

The "Submission Confirmation - Please confirm your data submission:" section contains the following fields:

- Grader Category: JFACC
- Grader User ID: SIMMONSB
- First Name: Bill
- Last Name: Simmons
- Rank: LCDR
- Command: USS George Washington

Bottom Screenshot: Submission Confirmation

The browser window title is "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.120.179.192/ITAPSAdmin/". The page header displays the iTAPS logo and "Integrated Theater Assessment Profiling System".

The left sidebar is identical to the top screenshot.

The main content area shows the "Submission Confirmation" section with the following fields:

- Command: USS George Washington
- Department: Air Wing
- Office Phone: 555-5555
- Commercial Phone: 555-555-5555
- Cell Phone: 555-555-5555
- Pager: 555-555-5555
- Email Address: simmonsbg@gw.navy.mil

At the bottom of the form are two buttons: "Continue" and "Cancel".

Figure 59. Confirming your entry

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 59). If you made a mistake or do not wish to create this Element Title entry, Click "Cancel". You will be returned to the previous screen. If the entry is correct, click "Continue" and ITAPS will update its database.

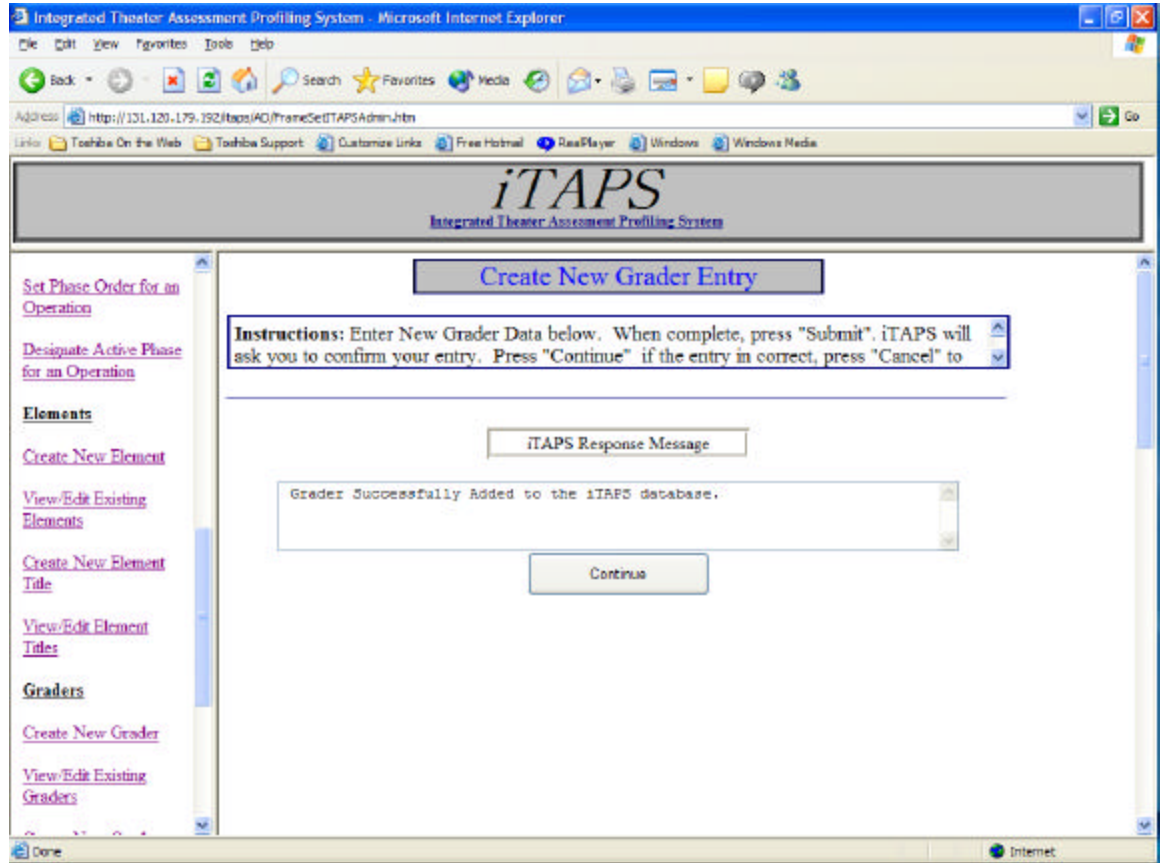


Figure 60. iTAPS Responds

iTAPS will display a response message concerning your submission (see figure 60). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.13 Editing/Deleting Graders

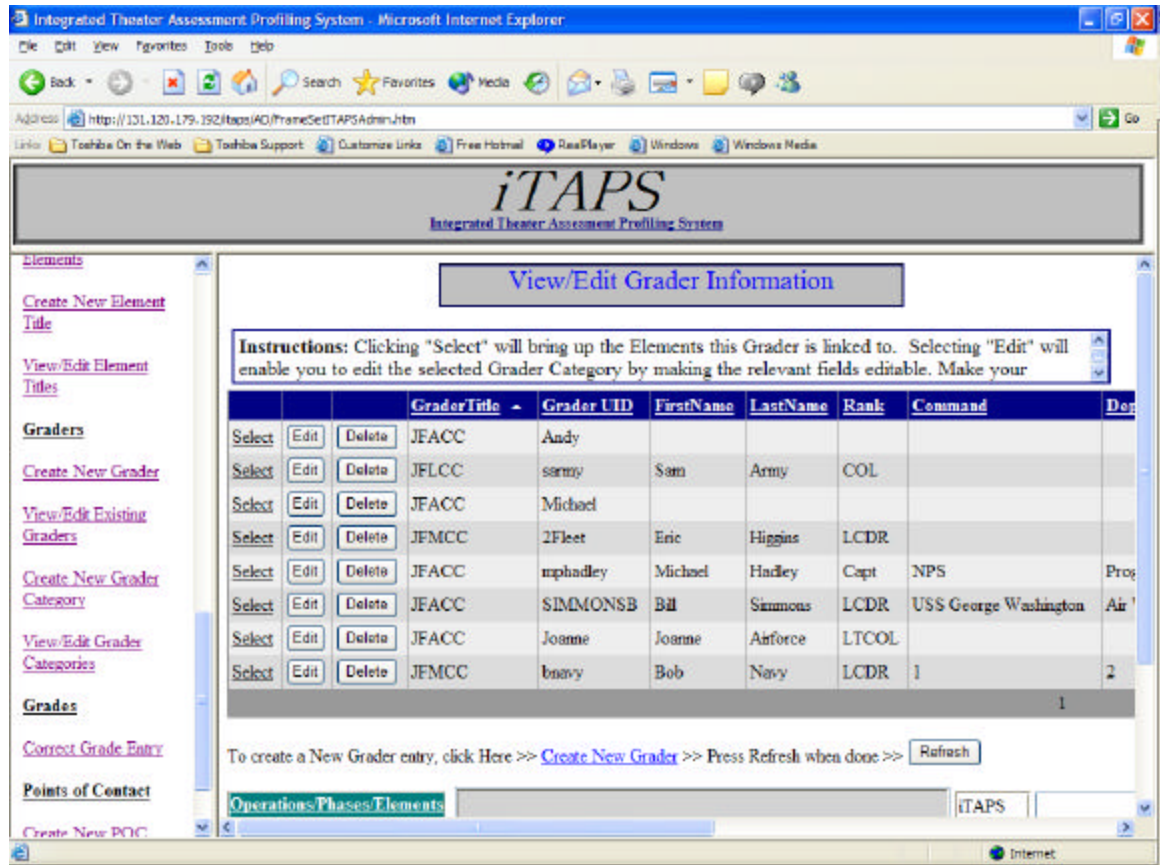


Figure 61. Viewing and Editing Graders

To view, edit or delete Graders already in ITAPS select “View/Edit Graders” from the Admin Tools menu (see figure 61). The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Grader, click the “Create New Grader” hyperlink. When you are done creating the new Grader, click “Refresh” to update the datagrid with your entry.

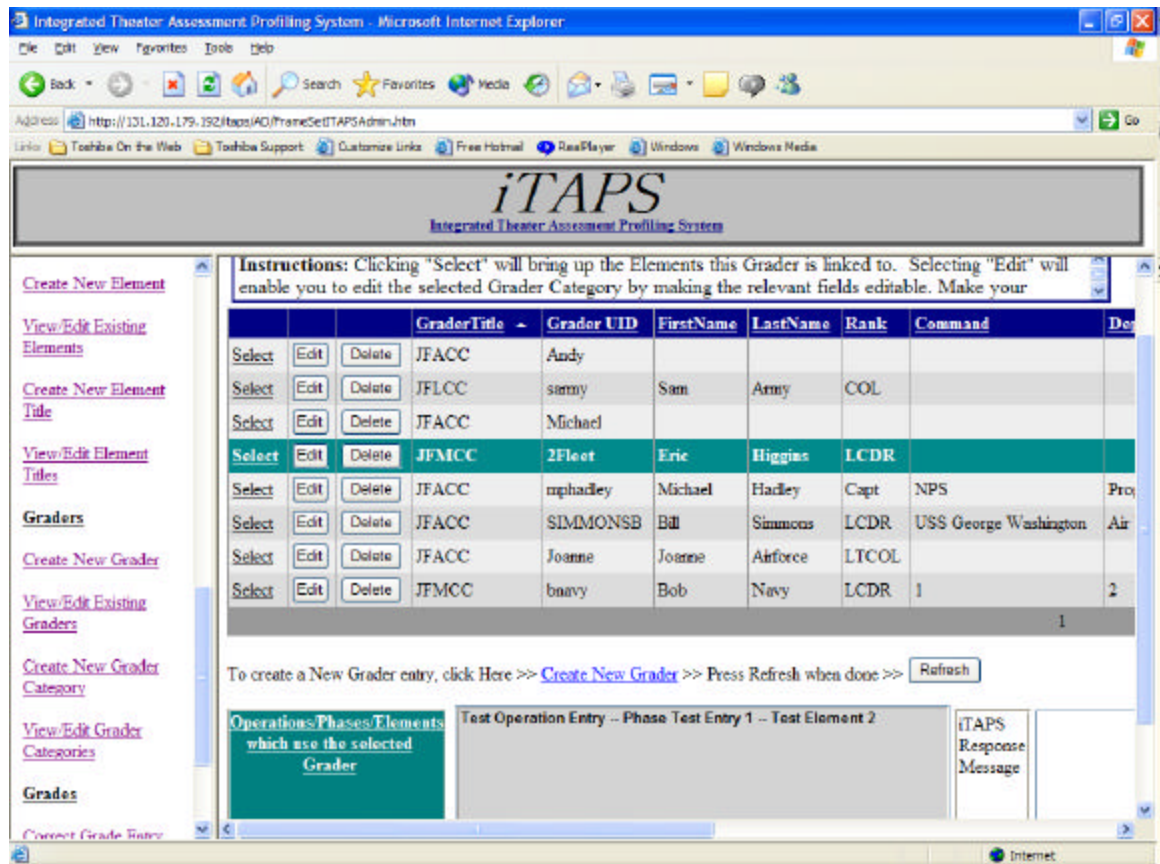


Figure 62. Selecting a Grader

Elements that use the selected Grader will appear in the list box at the bottom of the page. You cannot delete a Grader which is in use by an Element.

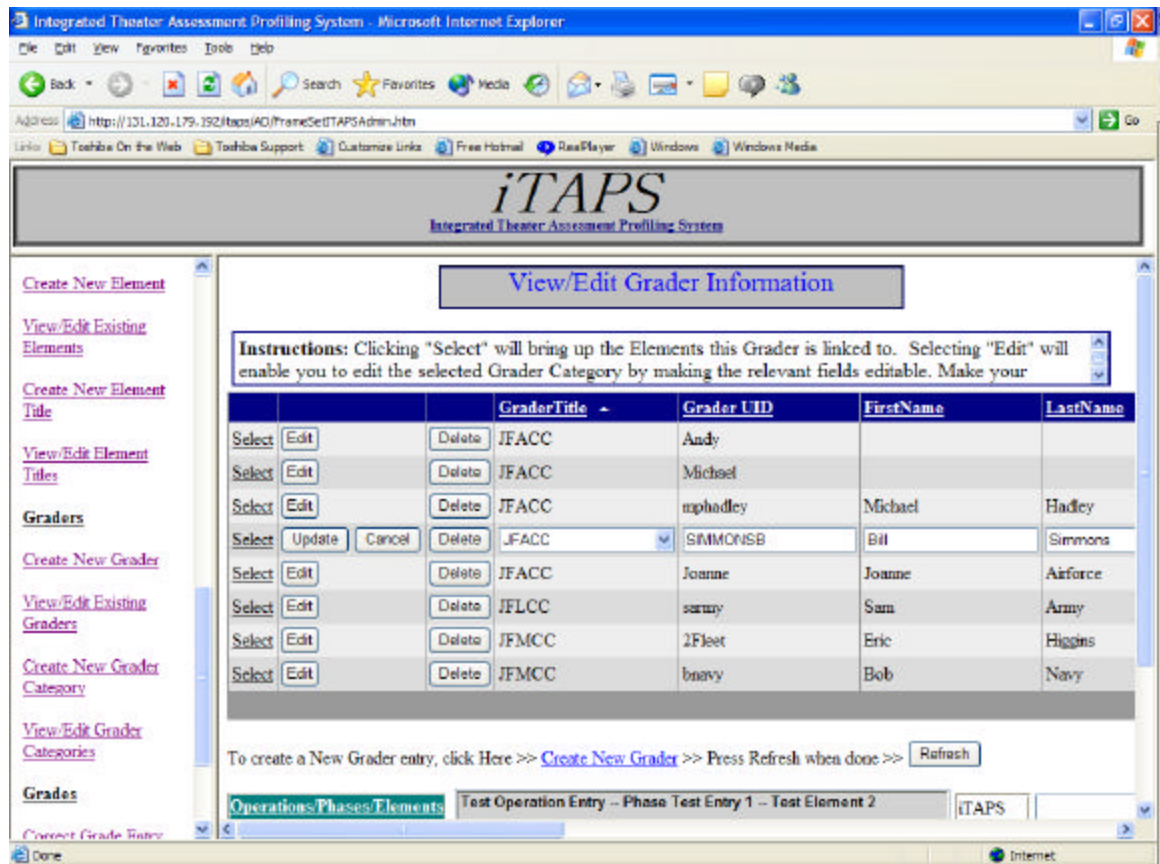


Figure 63. Editing a Grader

To Edit a Grader, click the "Edit" button for that Grader. An Edit Template will be displayed for that Grader (see figure 63). Make the changes you desire and click "Update" to enter your changes or "Cancel" to cancel your changes.

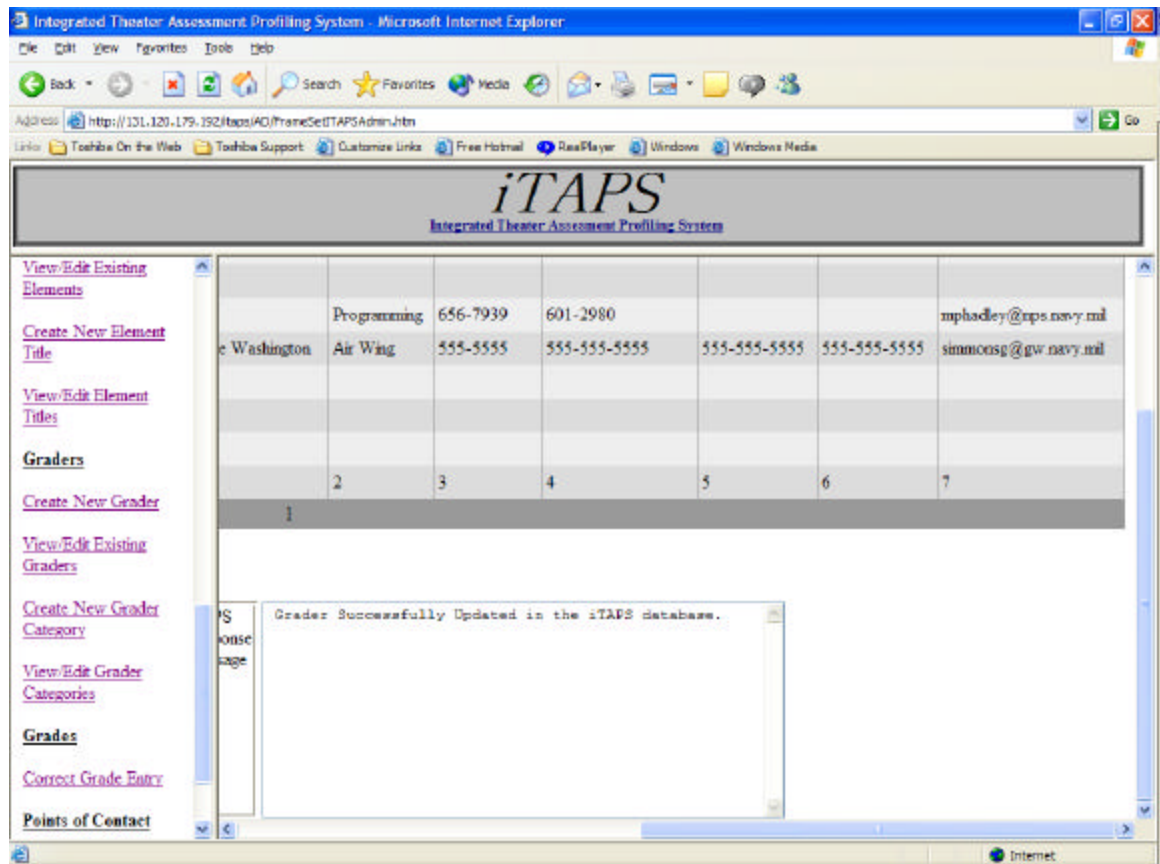


Figure 64. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 64). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

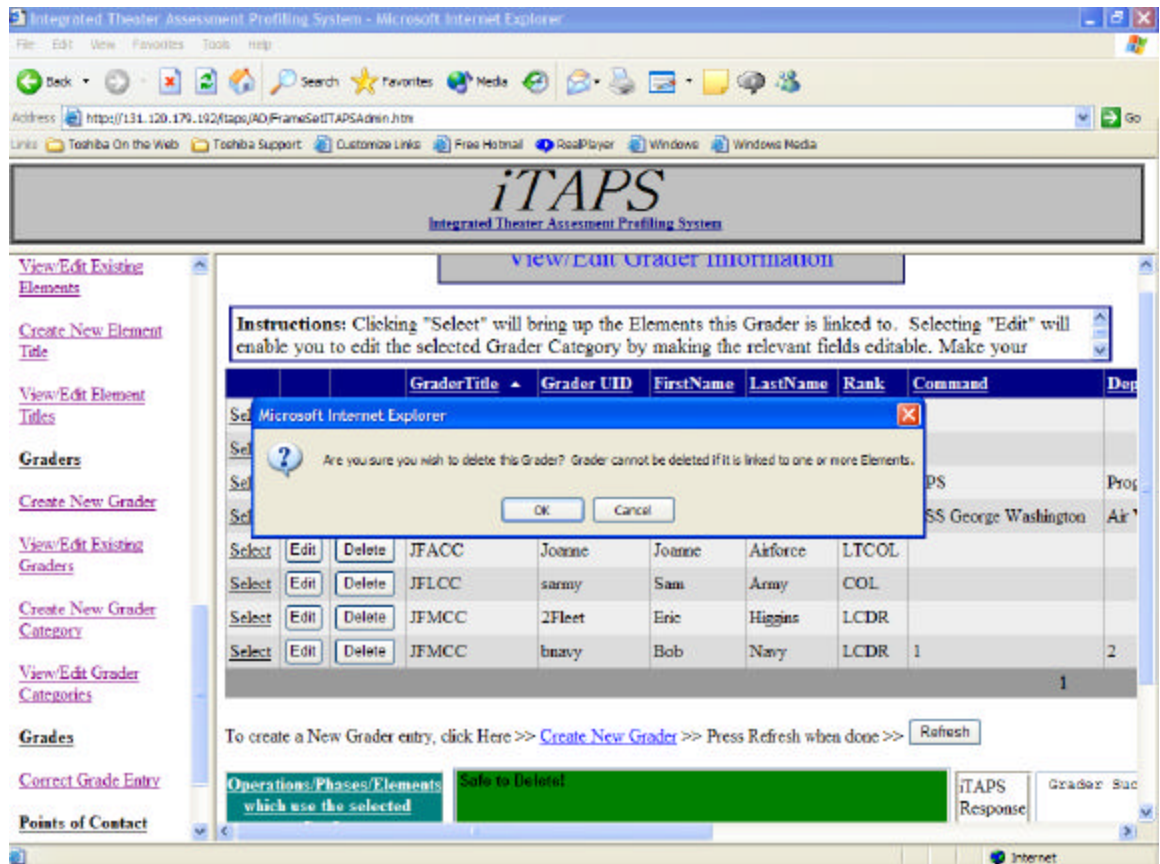


Figure 65. Deleting a Grader

To Delete a Grader, click the "Delete" button for the desired Grader. A confirmation box will ask you to confirm your selection (see figure 65). Clicking "OK" will cause the Grader to be permanently deleted if it is not being used by any Elements. Clicking "Cancel" will cancel your delete request. Graders in use by an Element cannot be deleted.

5.1.14 Creating Grader Categories

The screenshot shows a Microsoft Internet Explorer window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar displays "http://131.129.179.192/itaps/AD/FrameSetITAPSAdmin.htm". The page features a header with the "iTAPS" logo and the text "Integrated Theater Assessment Profiling System". On the left side, there is a vertical menu with links: "View/Edit Existing Elements", "Create New Element Title", "View/Edit Element Titles", "Graders", "Create New Grader", "View/Edit Existing Graders", "Create New Grader Category", "View/Edit Grader Categories", "Grades", and "Correct Grade Entry". The main content area is titled "Create New Grader Category" and contains an instruction box: "Instructions: Enter a New Grader Category and a Point of Contact(if desired)and press 'Submit'. iTAPS will ask you to confirm your entry. Press 'Continue' if the entry in". Below the instructions, there are two input fields: "New Grader Category:" with an empty text box, and "Point of Contact:" with a dropdown menu currently set to "None". A "Submit" button is located below these fields. The status bar at the bottom of the browser window shows "Done" and "Internet".

Figure 67. Creating a Grader Category

To create a new Grader Category, select "Create New Grader Category" from the Admin Tools menu (see figure 67). The Grader Category is required, the POC selection is not.

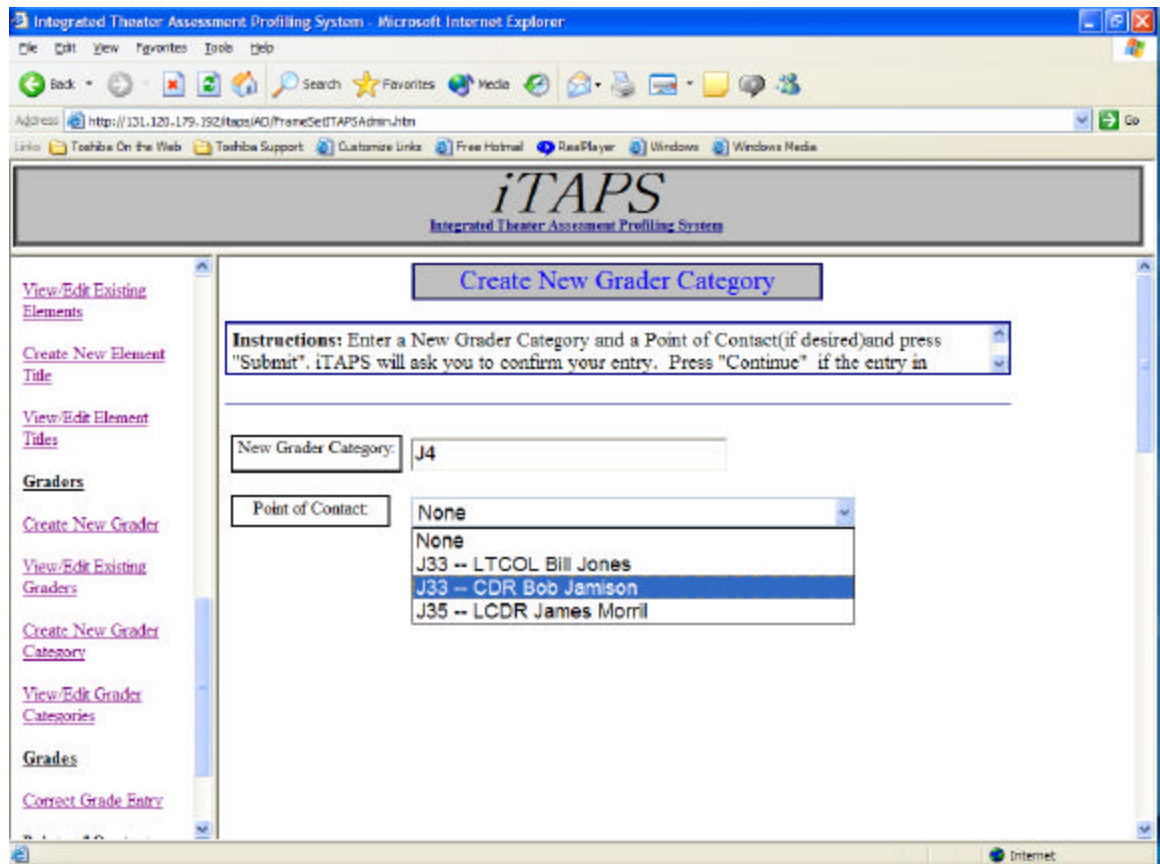


Figure 68. POC Selection

Use the Point of Contact Drop down list to select a POC if desired. Click "Submit" when satisfied with your entry.

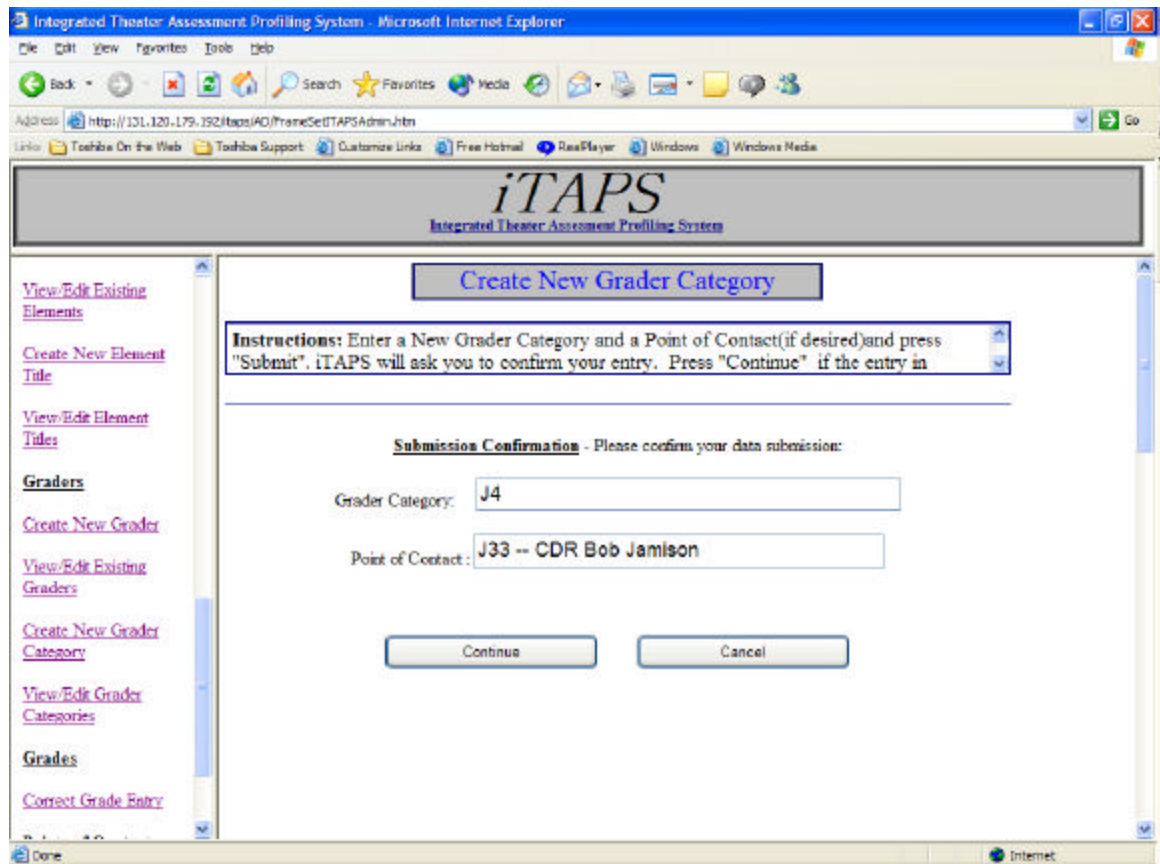


Figure 69. Submission Confirmation

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 69). If you made a mistake or do not wish to create this Grader Category entry, Click “Cancel”. You will be returned to the previous screen. If the entry is correct, click “Continue” and ITAPS will update its database.

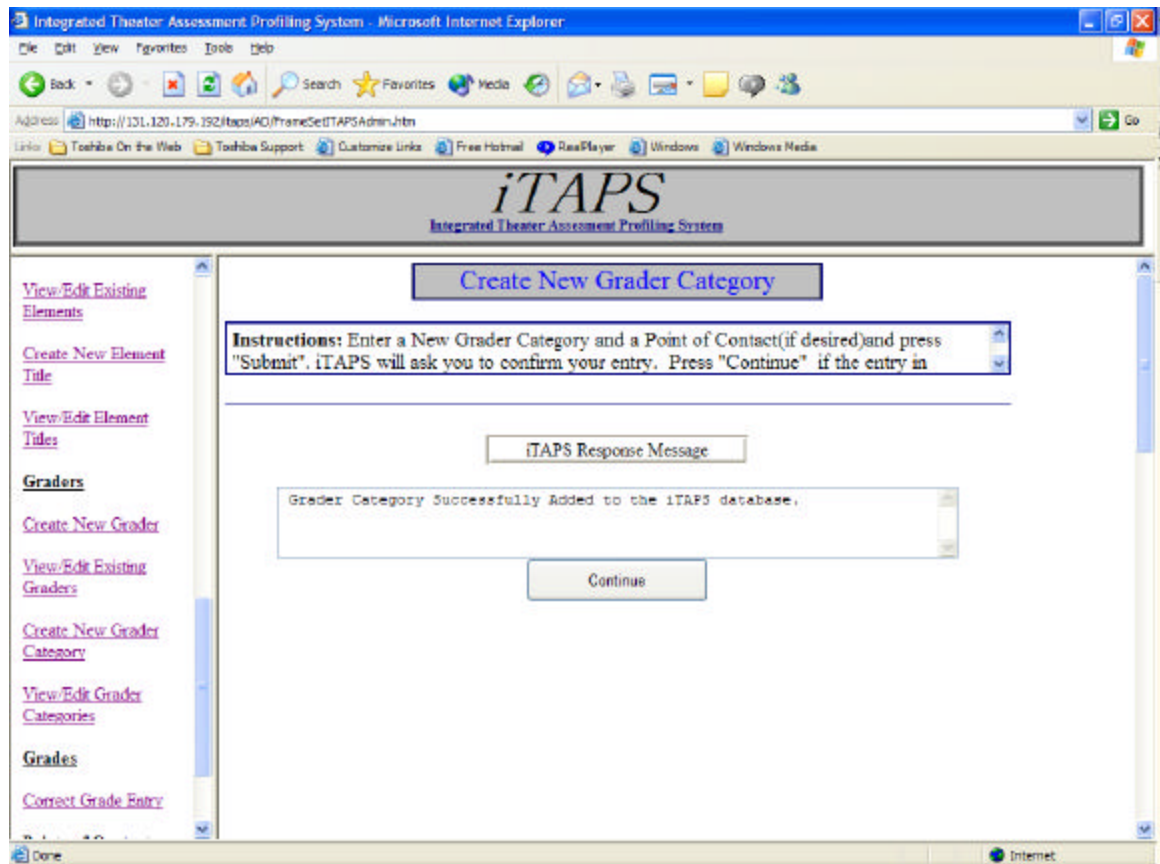


Figure 70. ITAPS Responds

ITAPS will display a response message concerning your submission (see figure 70). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.15 Editing/Deleting Grader Categories

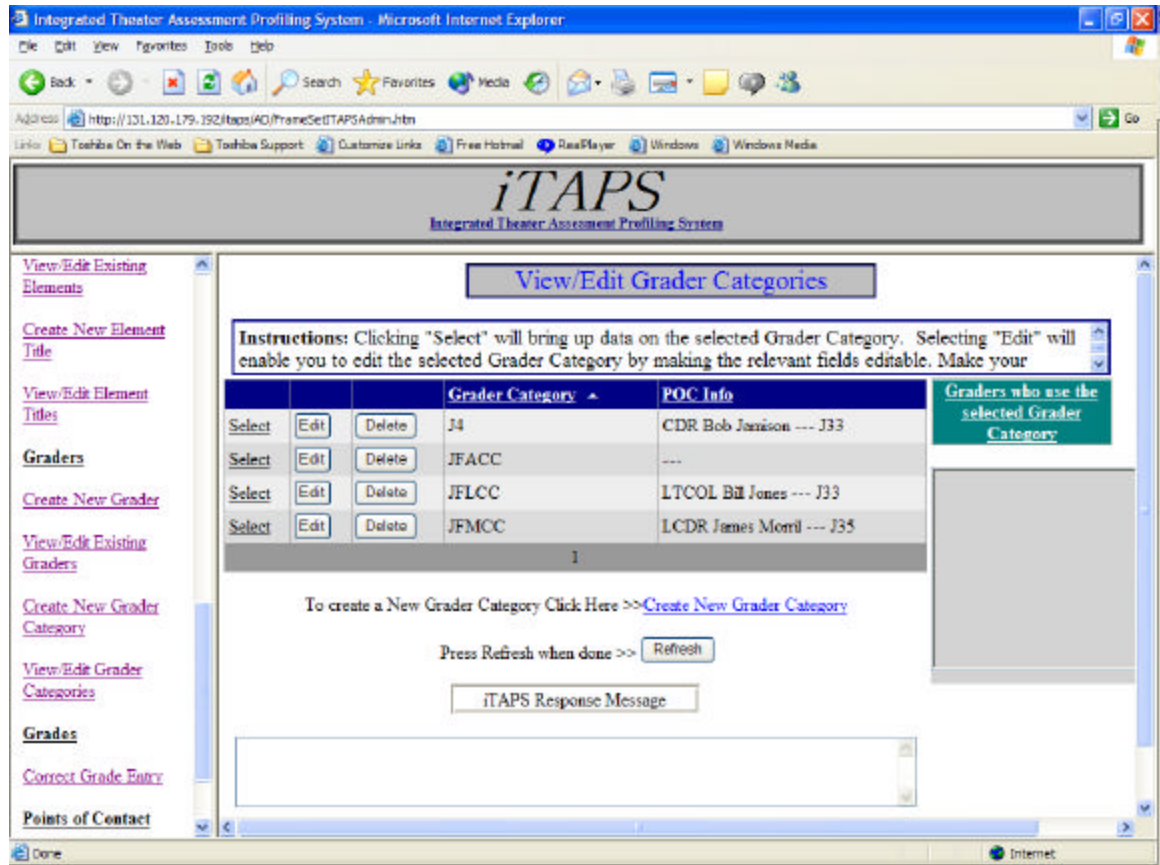


Figure 71. Viewing and Editing Grader Categories

To view, edit or delete Grader Categories already in ITAPS select "View/Edit Grader Categories" from the Admin Tools menu (see figure 71). The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new Grader Category, click the "Create New Grader Category" hyperlink. When you are done creating the new Grader Category, click "Refresh" to update the datagrid with your entry.

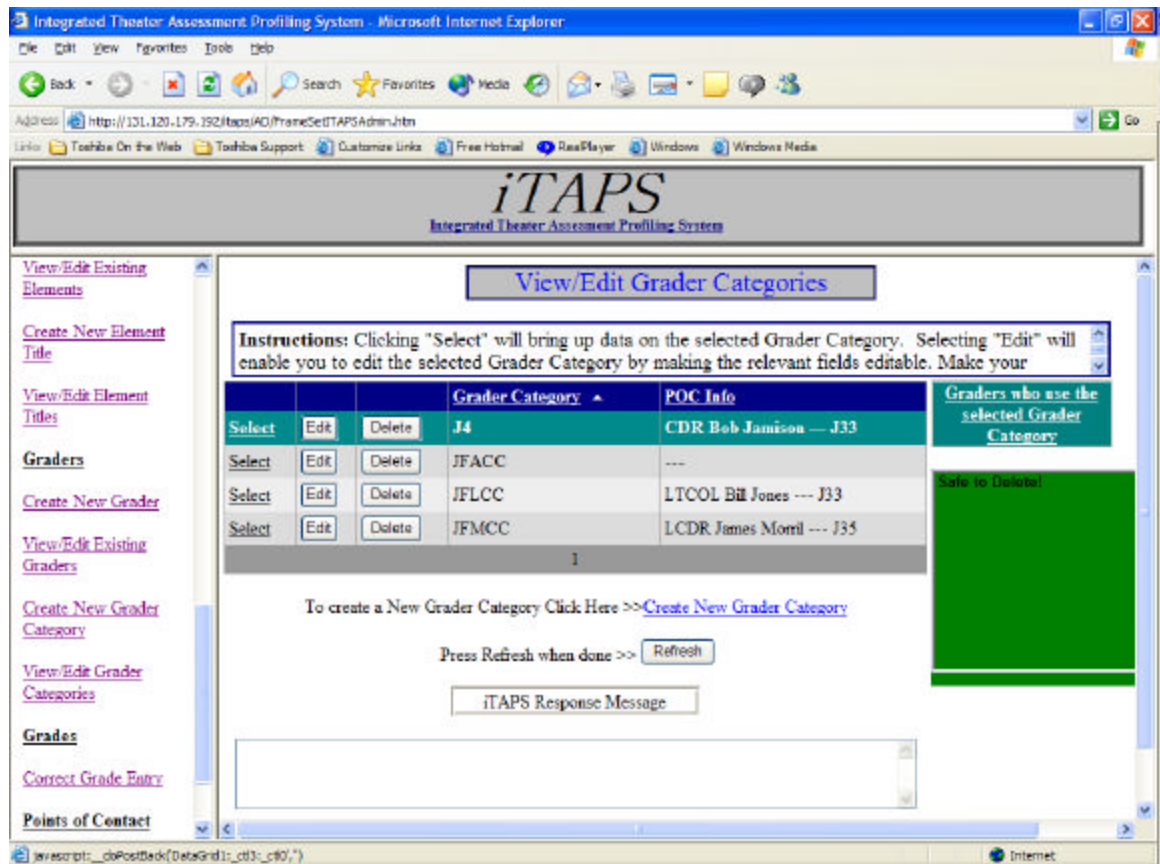


Figure 72. Selecting a Grader Category

Graders that use the selected Grader will appear in the list box to the right. You cannot delete a Grader Title which is in use by a Grader.

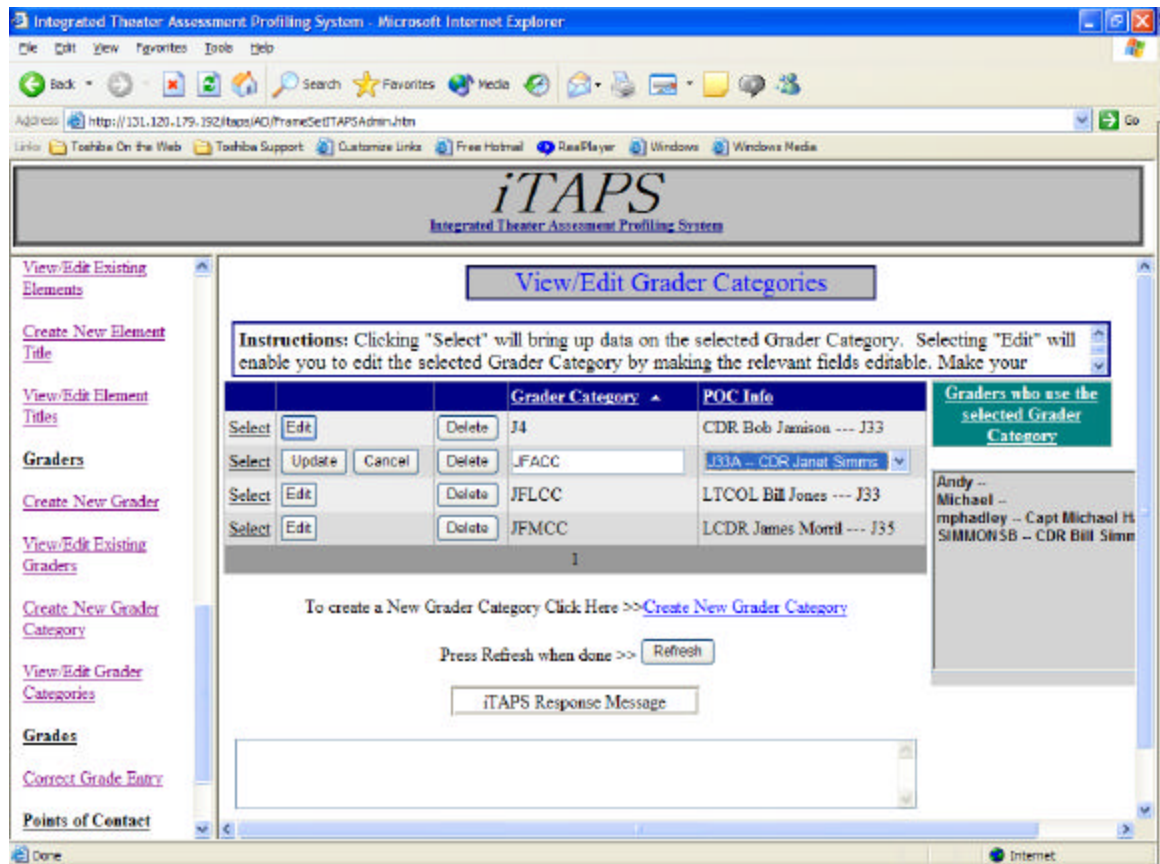


Figure 73. Editing a Grader

To Edit a Grader Category, click the "Edit" button for that Grader Category. An Edit Template will be displayed for that Grader Category (see figure 73). Make the changes you desire and click "Update" to enter your changes or "Cancel" to cancel your changes.

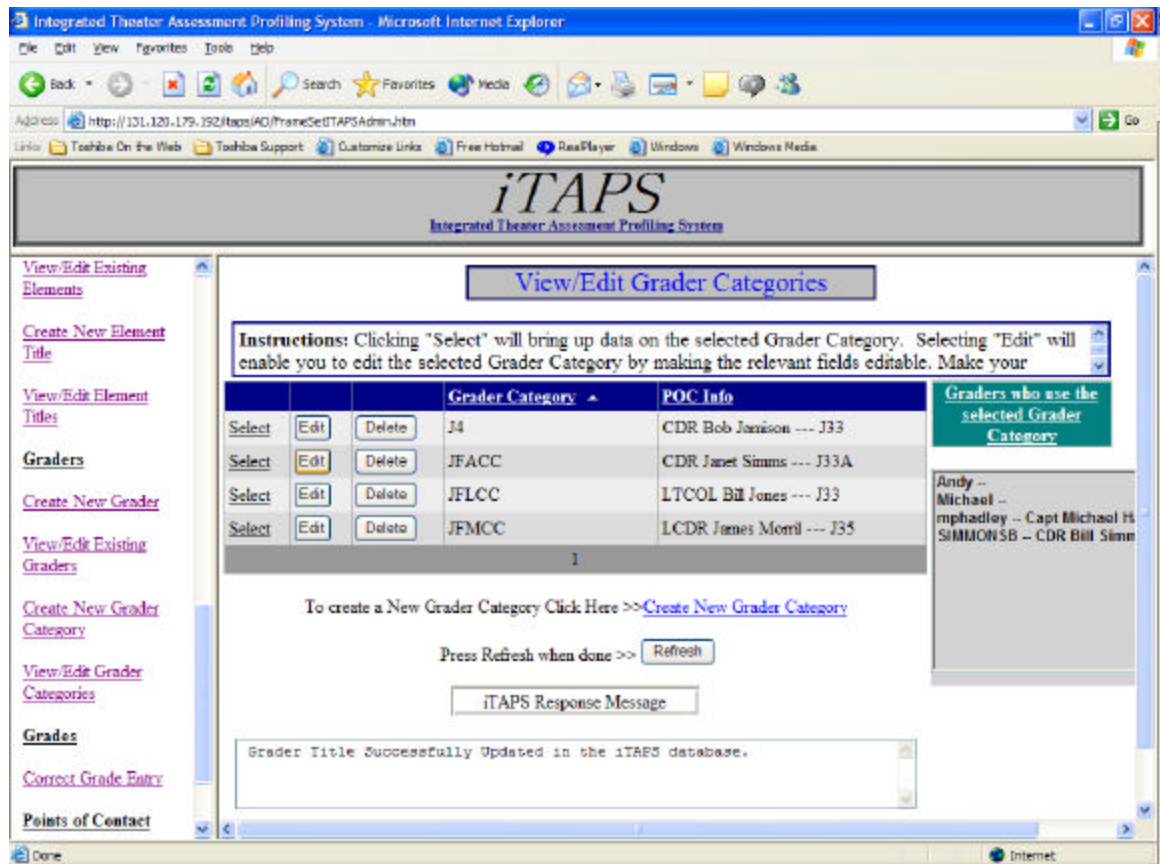


Figure 74. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 74). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

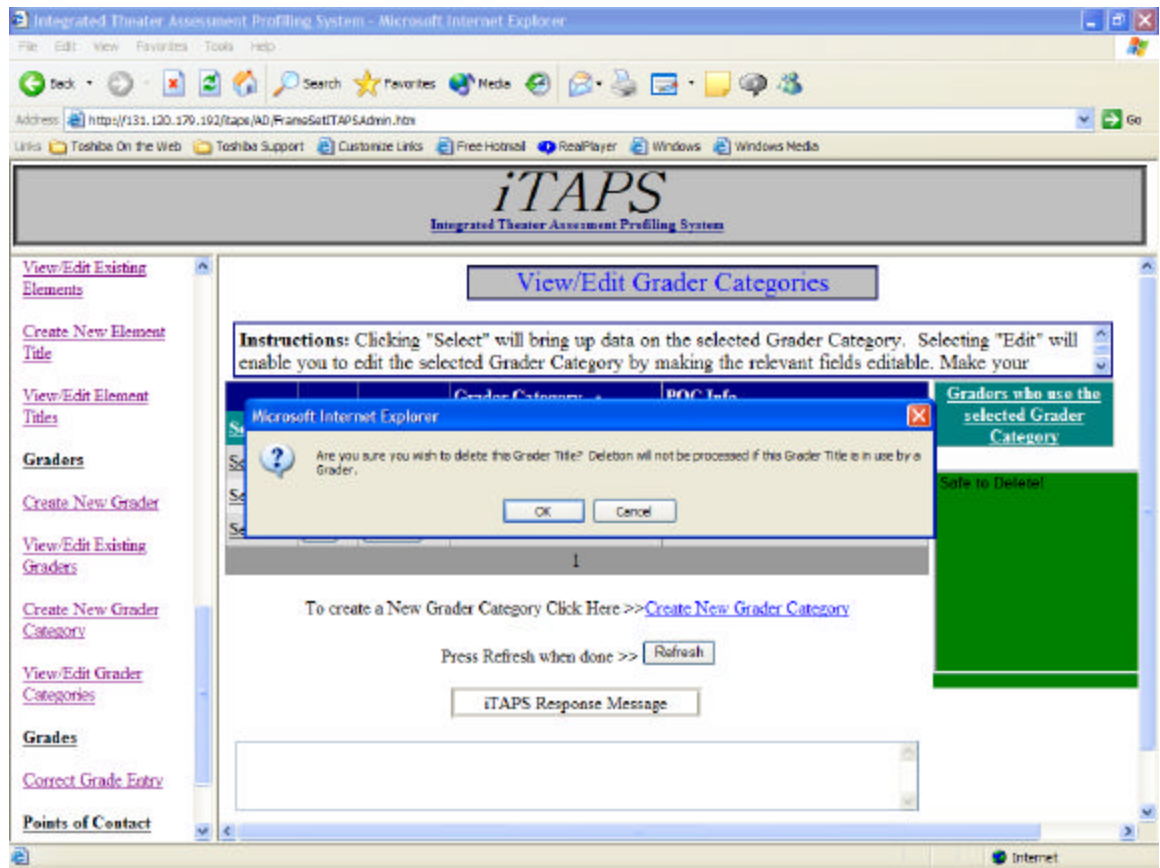


Figure 75. Deleting a Grader

To Delete a Grader Category, click the “Delete” button for the desired Grader Category. A confirmation box will ask you to confirm your selection (see figure 75). Clicking “OK” will cause the Grader Category to be permanently deleted if it is not being used by any Graders. Clicking “Cancel” will cancel your delete request. Grader Categories in use by an Grader cannot be deleted.

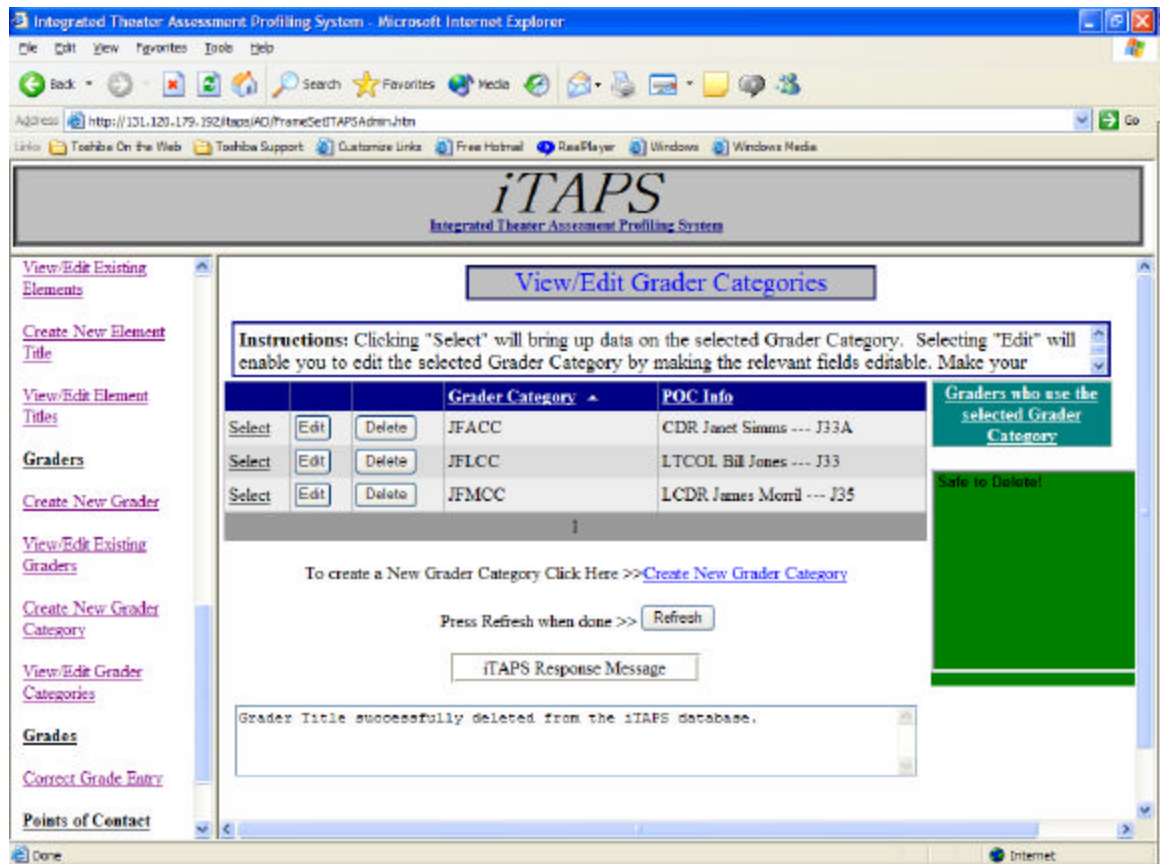


Figure 76. ITAPS Responds

After the deletion, ITAPS will list the results of the operation in the Response Message window (see figure 76). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.16 Creating Point-of-Contact

The screenshot shows a Microsoft Internet Explorer window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar displays "http://131.129.179.192/itaps/AD/FrameSetITAPSAdmin.htm". The page features a header with the "iTAPS" logo and the text "Integrated Theater Assessment Profiling System". A left-hand navigation menu includes links such as "View/Edit Element Titles", "Graders", "Create New Grader", "View/Edit Existing Graders", "Create New Grader Category", "View/Edit Grader Categories", "Grades", "Correct Grade Entry", "Points of Contact", "Create New POC", and "View/Edit Existing POC". The main content area is titled "Create New Point of Contact" and contains an instruction box: "Instructions: Enter New Point of Contact Data below. When complete, press 'Submit'. iTAPS will ask you to confirm your entry. Press 'Continue' if the entry in". Below this is a section titled "Point of Contact Information" with six input fields: "Title:", "First Name:", "Last Name:", "Rank:", "Command:", and "Department:". The status bar at the bottom shows "Done" and "Internet".

Figure 77. Creating a POC

To create a new POC, select "Create New POC" from the Admin Tools menu (see figure 77). The POC Title is required, all other fields are not.

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

Address: http://131.120.179.192/itaps/NOFrameSetITAPSAdmin.htm

iTAPS
Integrated Theater Assessment Profiling System

Create New Point of Contact

Instructions: Enter New Point of Contact Data below. When complete, press "Submit". iTAPS will ask you to confirm your entry. Press "Continue" if the entry in

Point of Contact Information

Title: J33A

First Name: Janet

Last Name: Simms

Rank: CDR

Command: COMSECONDFLT

Department: Current Ops

Office Phone: 555-5555

Commercial Phone: 555-555-5555

Cell Phone: 555-555-5555

Pager: 555-555-5555

Email Address: simmsj@secondflt.navy.mil

Submit

Figure 78. Creating a POC

Click the "Submit" button when satisfied with your entry.

The figure consists of two screenshots of the iTAPS web application interface, showing the submission confirmation process.

Top Screenshot: Create New Point of Contact

The browser window title is "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.120.179.132/itaps/POCFrameSetITAPSAdmin.htm". The iTAPS logo is at the top. The left sidebar contains links: View/Edit Element Titles, Graders, Create New Grader, View/Edit Existing Graders, Create New Grader Category, View/Edit Grader Categories, Grades, Correct Grade Entry, Points of Contact, Create New POC, and View/Edit Existing POC.

The main content area has a heading "Create New Point of Contact". Below it is an instruction box: "Instructions: Enter New Point of Contact Data below. When complete, press 'Submit'. iTAPS will ask you to confirm your entry. Press 'Continue' if the entry is".

Below the instruction box is a "Submission Confirmation - Please confirm your data submission:" section. It contains the following fields:

Title:	J33A
First Name:	Janet
Last Name:	Simms
Rank:	CDR
Command:	COMSECONDFLT
Department:	Current Ops

Bottom Screenshot: Submission Confirmation

The browser window title is "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows "http://131.120.179.132/itaps/POCFrameSetITAPSAdmin.htm". The iTAPS logo is at the top. The left sidebar contains the same links as the top screenshot.

The main content area has a heading "Submission Confirmation". It contains the following fields:

Office Phone:	555-5555
Commercial Phone:	555-555-5555
Cell Phone:	555-555-5555
Pager:	555-555-5555
Email Address:	simmsj@secondflt.navy.mil

At the bottom of the form are two buttons: "Continue" and "Cancel".

Figure 79. Submission Confirmation

ITAPS will now confirm your submission by listing your input and asking if you wish to continue or cancel your request (see figure 79). If you made a mistake or do not wish to create this POC entry, Click "Cancel". You will be returned to the previous screen. If the entry is correct, click "Continue" and ITAPS will update its database.

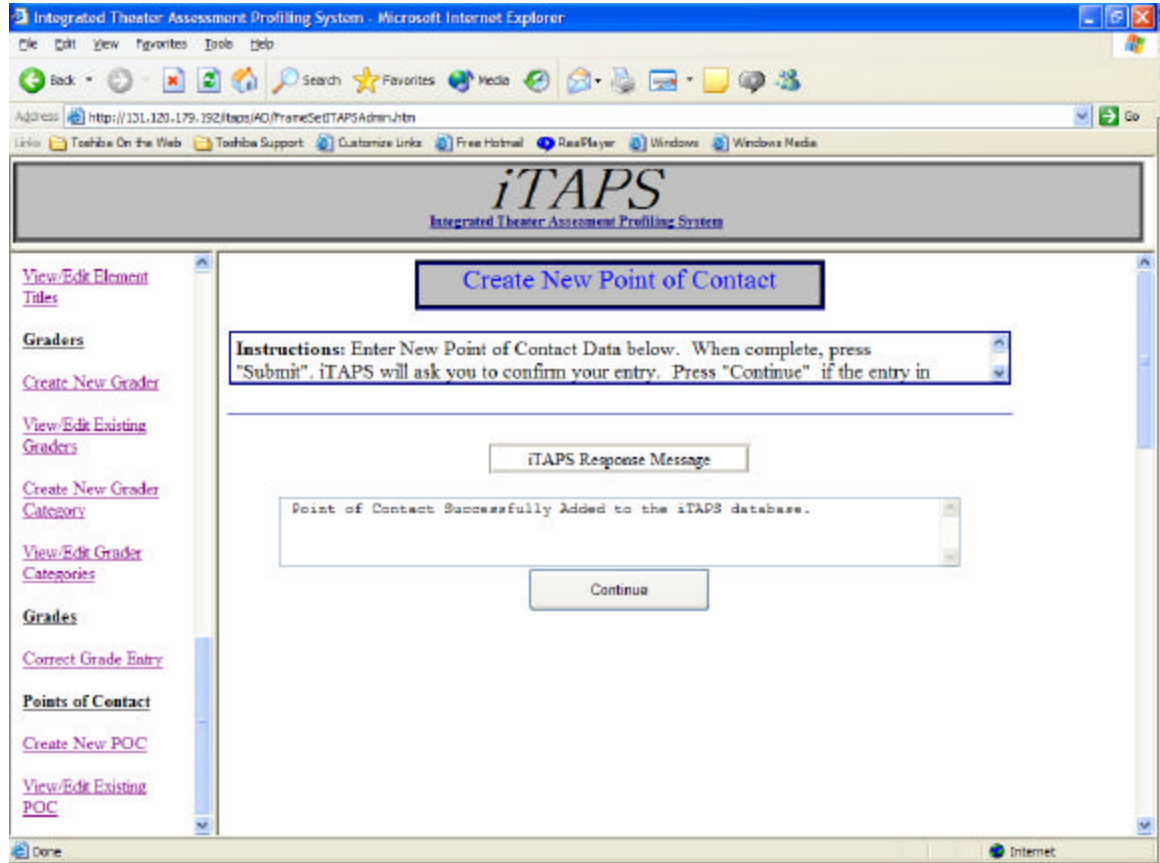


Figure 80. iTAPS Responds

iTAPS will display a response message concerning your submission (see figure 80). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.17 Editing/Deleting Points-of-Contact

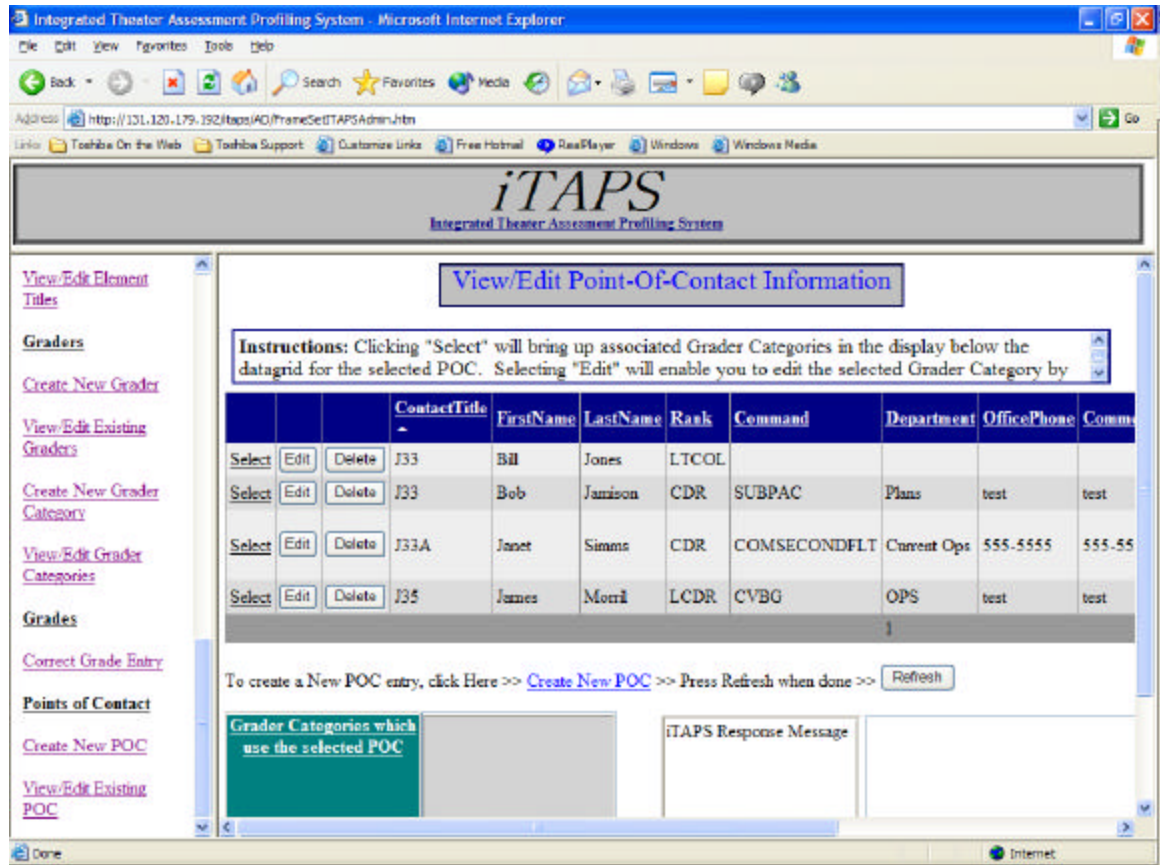


Figure 81. Viewing and Editing POCs

To view, edit or delete POCs already in ITAPS select "View/Edit Existing POC" from the Admin Tools menu (see figure 81). The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid. To create a new POC, click the "Create New POC" hyperlink. When you are done creating the new POC, click "Refresh" to update the datagrid with your entry.

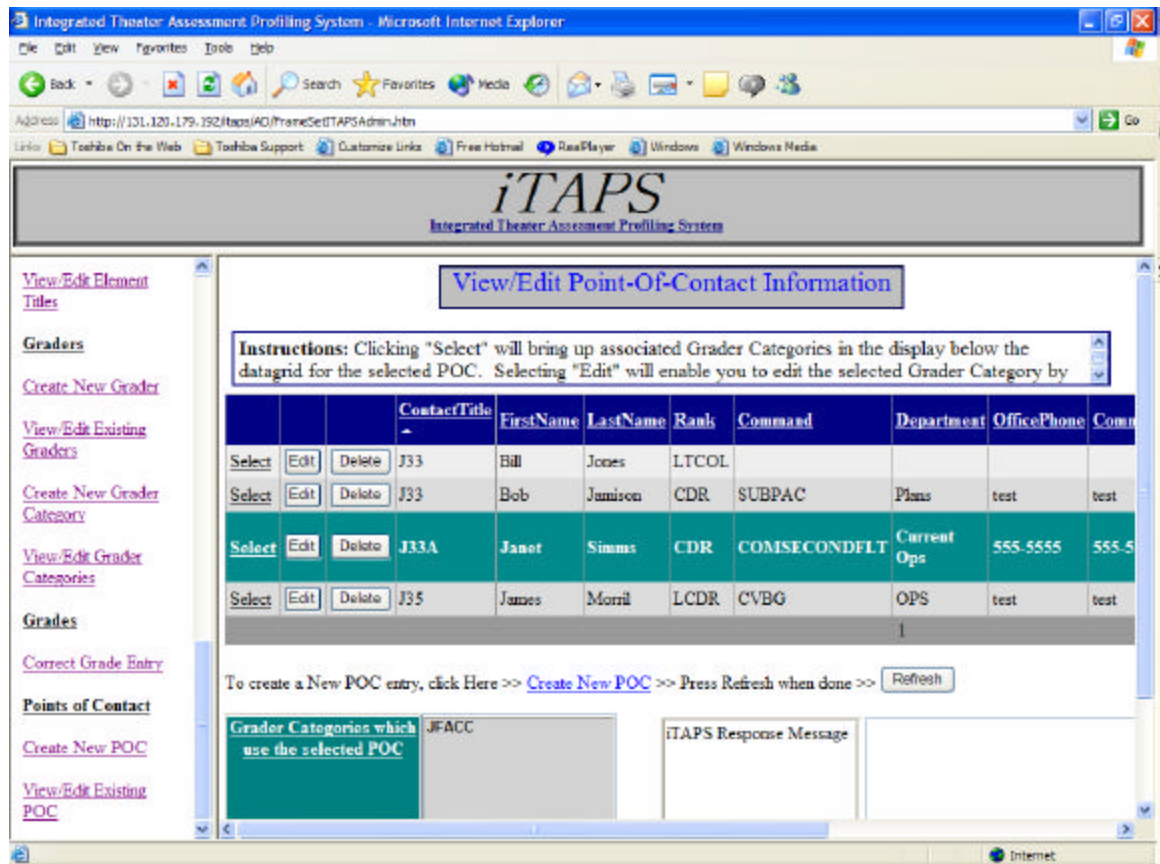


Figure 82. Selecting a POC

When a POC is selected, Grader Categories that use the selected POC will appear in the list box toward the bottom of the page.

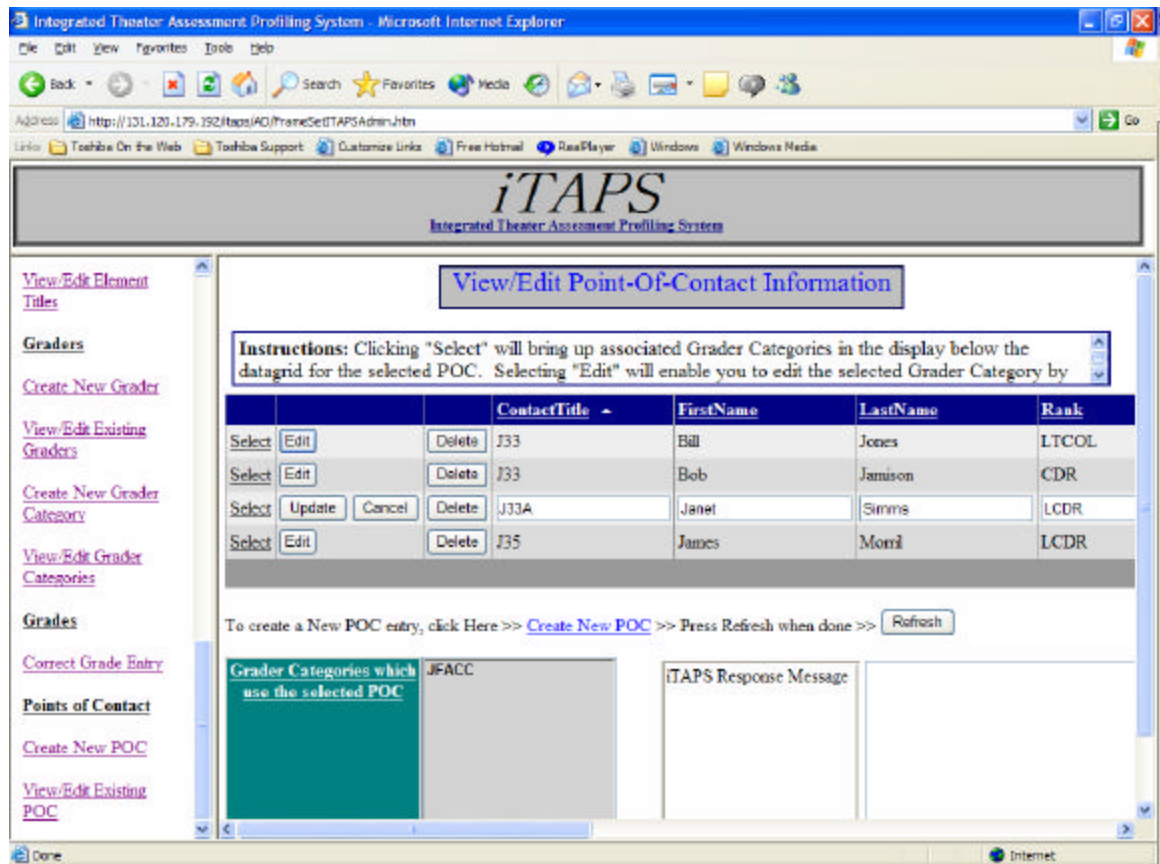


Figure 83. Editing a POC

To Edit a POC, click the "Edit" button for that POC. An Edit Template will be displayed for that POC (see figure 83). Make the changes you desire and click "Update" to enter your changes or "Cancel" to cancel your changes.

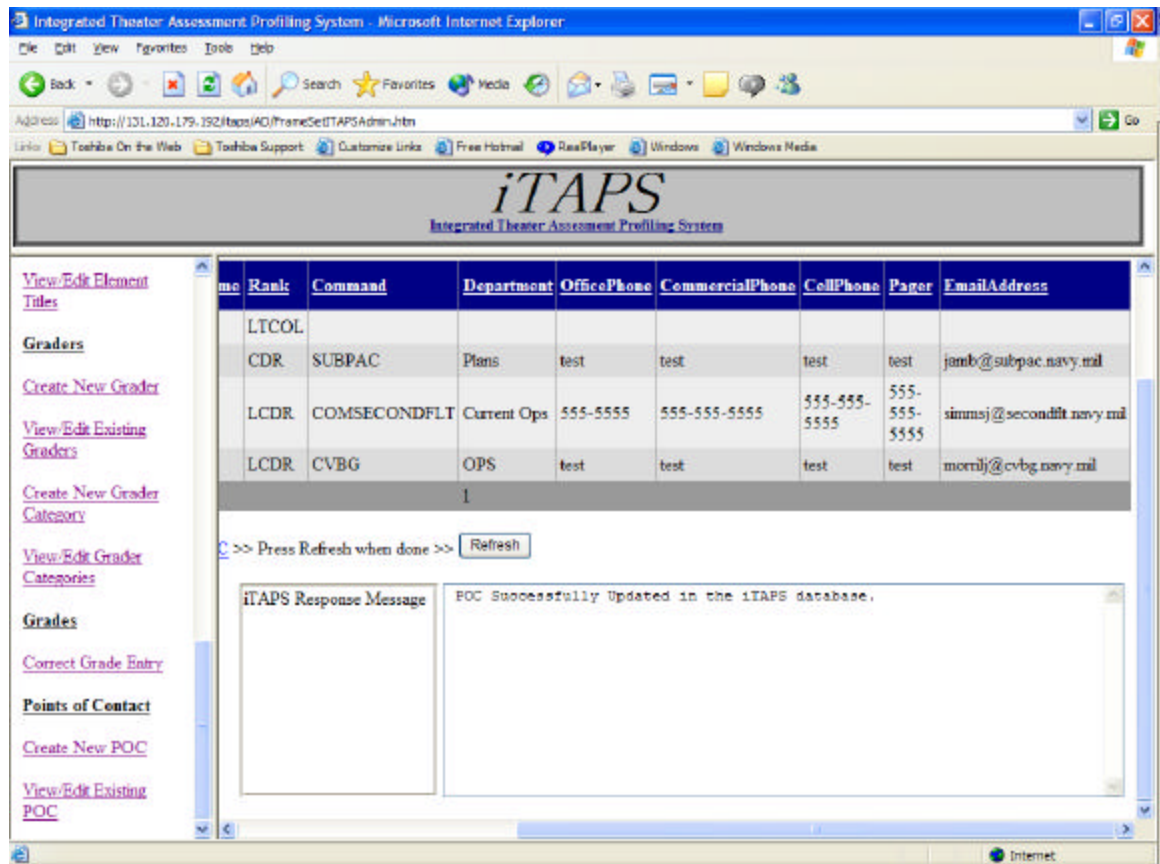


Figure 84. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 84). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

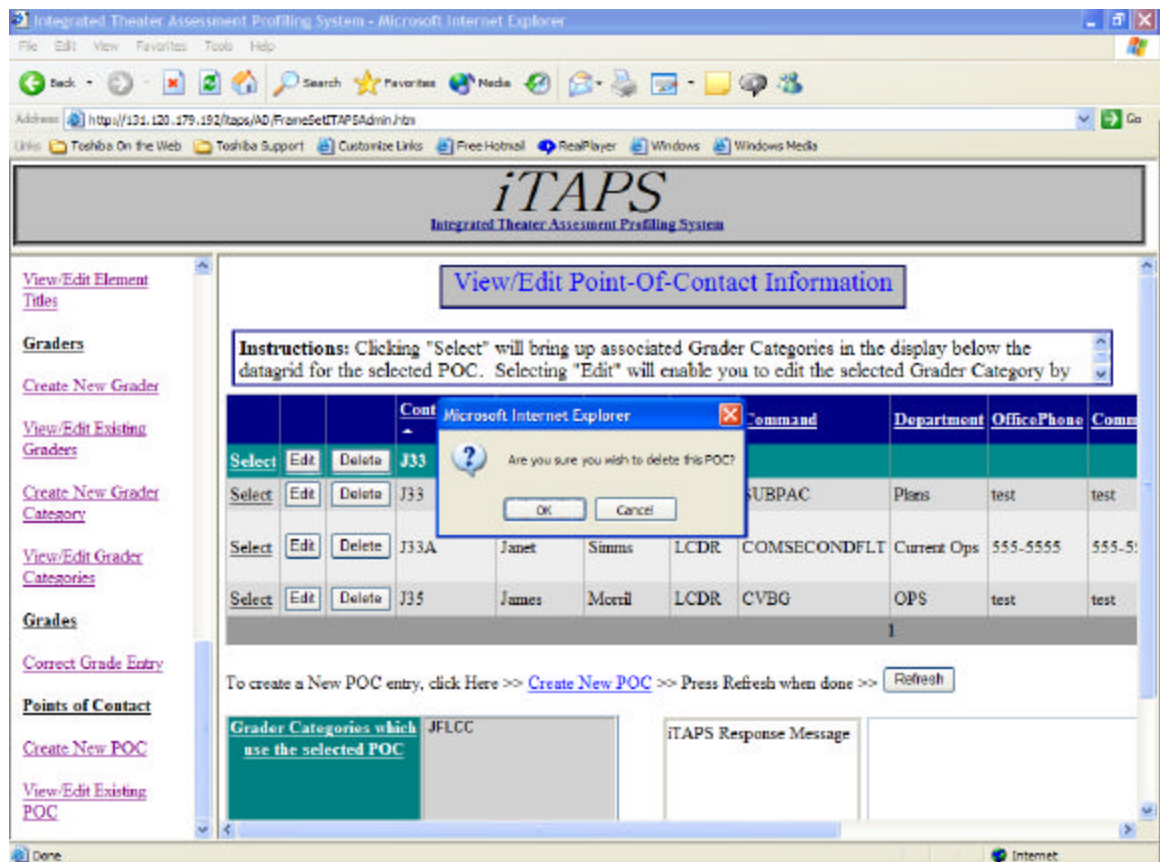


Figure 85. Deleting a POC

To Delete a POC, click the “Delete” button for the desired POC. A confirmation box will ask you to confirm your selection (see figure 85). Clicking “OK” will cause the POC to be permanently deleted. Clicking “Cancel” will cancel your delete request.

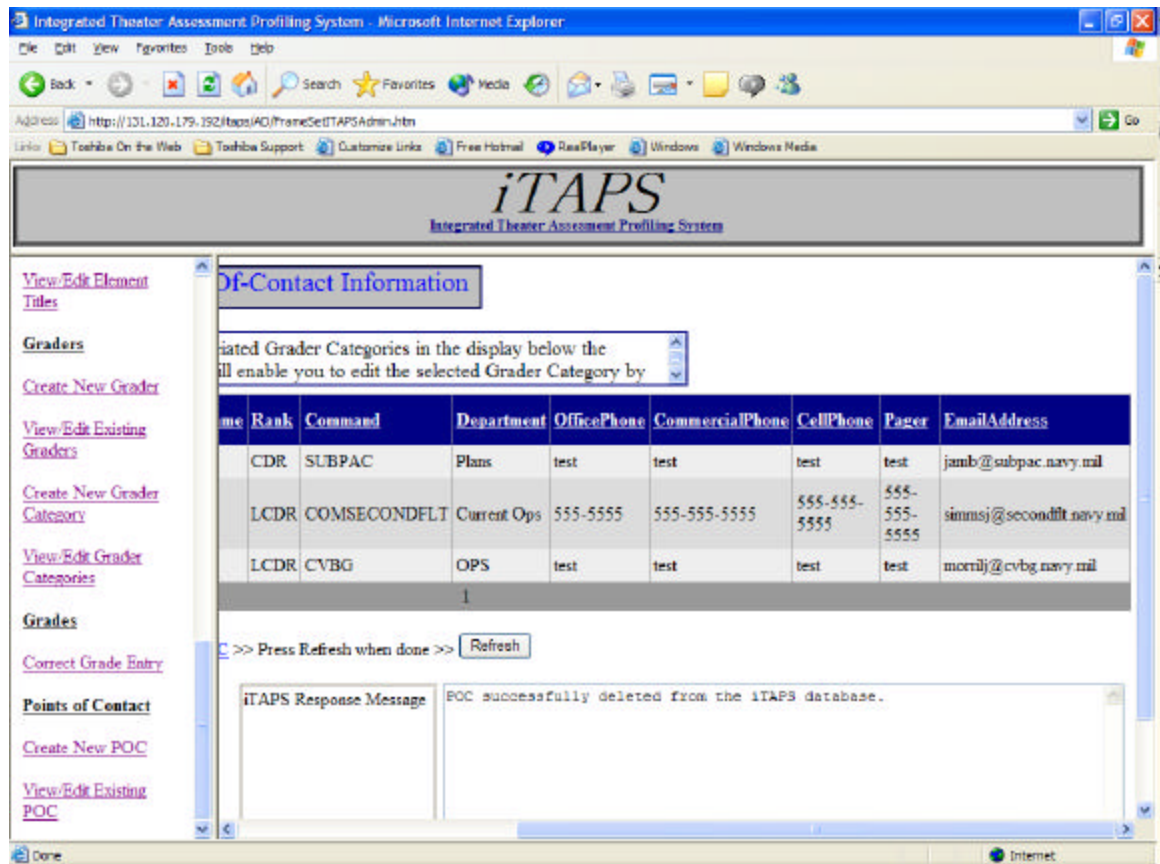


Figure 86. ITAPS Responds

After the deletion, ITAPS will list the results of the operation in the Response Message window (see figure 86). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.18 Editing/Deleting Grade Entries

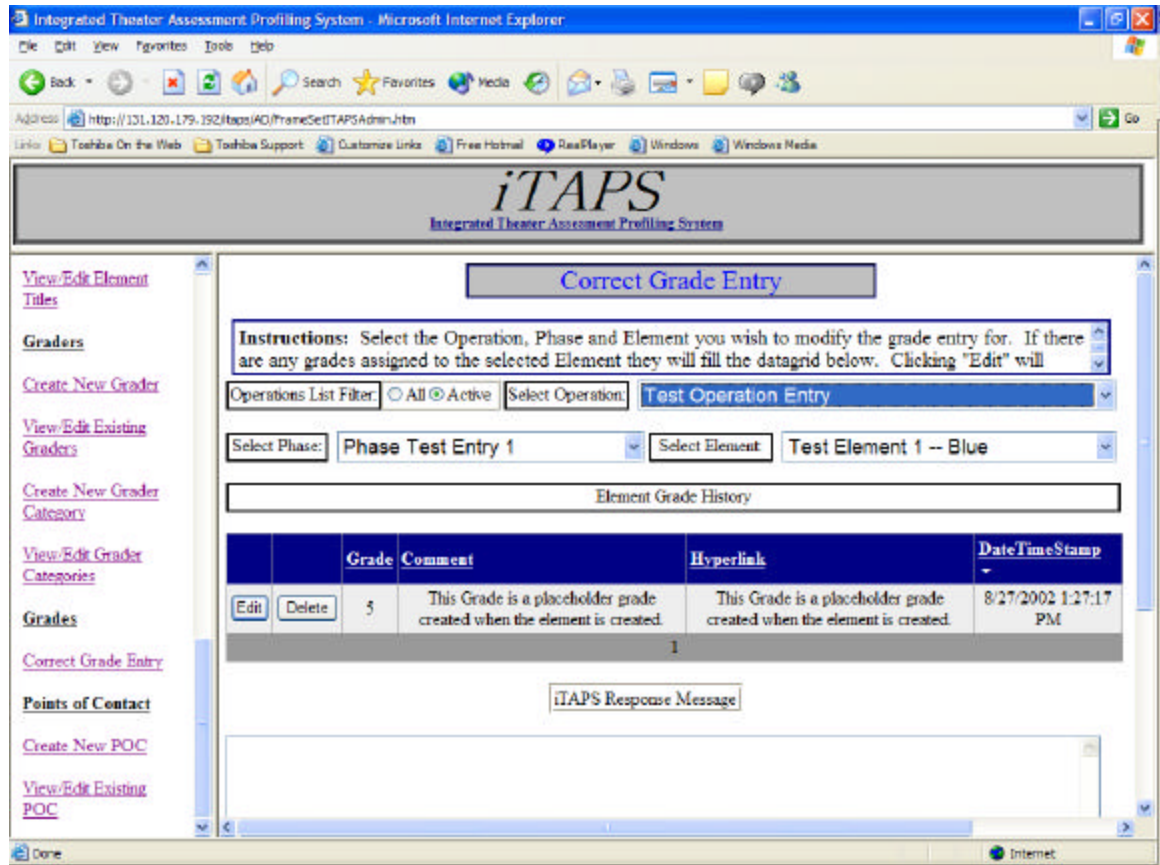


Figure 87. Editing and Deleting Grades

To view, edit or delete Grades already in ITAPS select "Correct Grade Entry" from the Admin Tools menu (see figure 87). Use the Operation, Phase and Element selection drop down lists to choose the Element you wish to edit grades for. The various data columns may be sorted in an ascending or descending manner by clicking on the respective column header. Up or down arrows next to the column header indicate the current sorting choice for the table. Pagination is provided by the page numbers along the bottom of the grid.

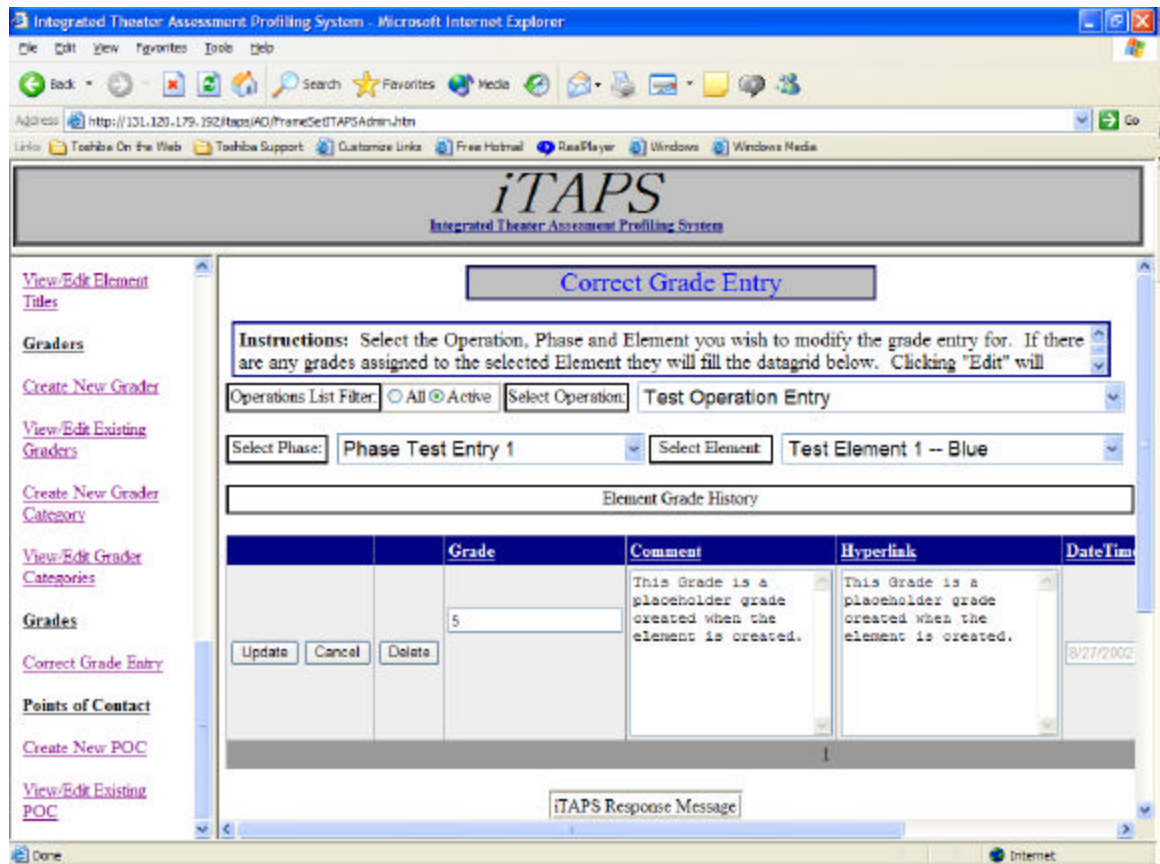


Figure 88. Editing a Grade

To Edit a Grade, click the "Edit" button for that Grade. An Edit Template will be displayed for that Grade (see figure 88). Make the changes you desire and click "Update" to enter your changes or "Cancel" to cancel your changes.

Some fields are not editable.

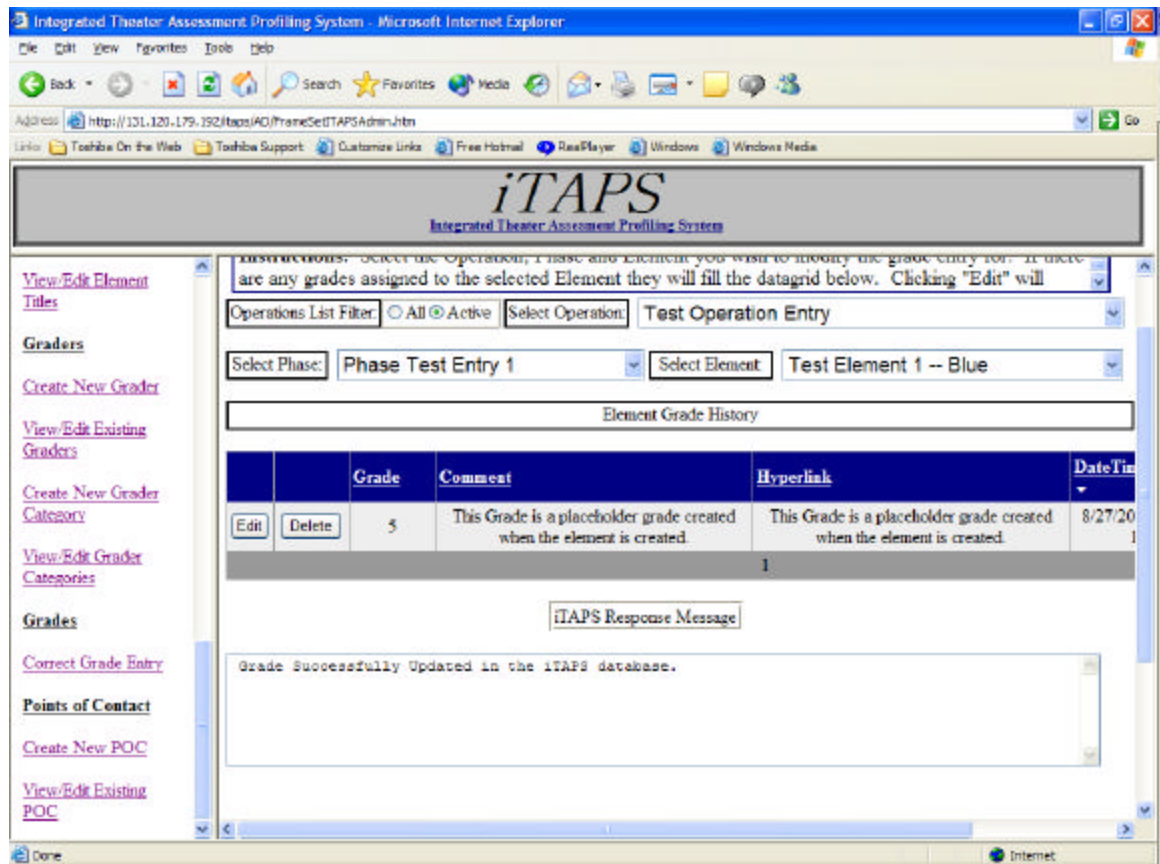


Figure 89. ITAPS Responds

After an update, ITAPS will list the results of the operation in the Response Message window (see figure 89). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

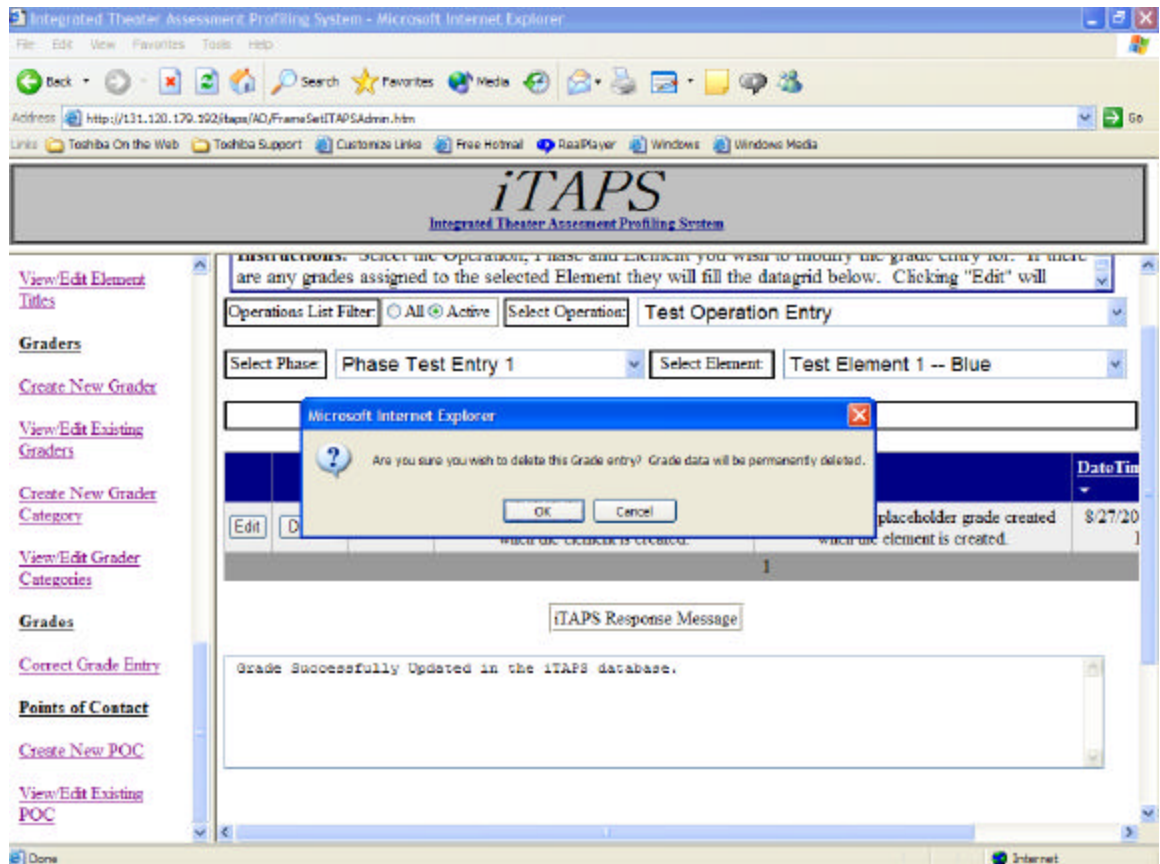


Figure 90. Deleting a Grade

To Delete a Grade, click the “Delete” button for the desired Grade. A confirmation box will ask you to confirm your selection (see figure 90). Clicking “OK” will cause the POC to be permanently deleted. Clicking “Cancel” will cancel your delete request. You cannot delete the last Grade for an Element.

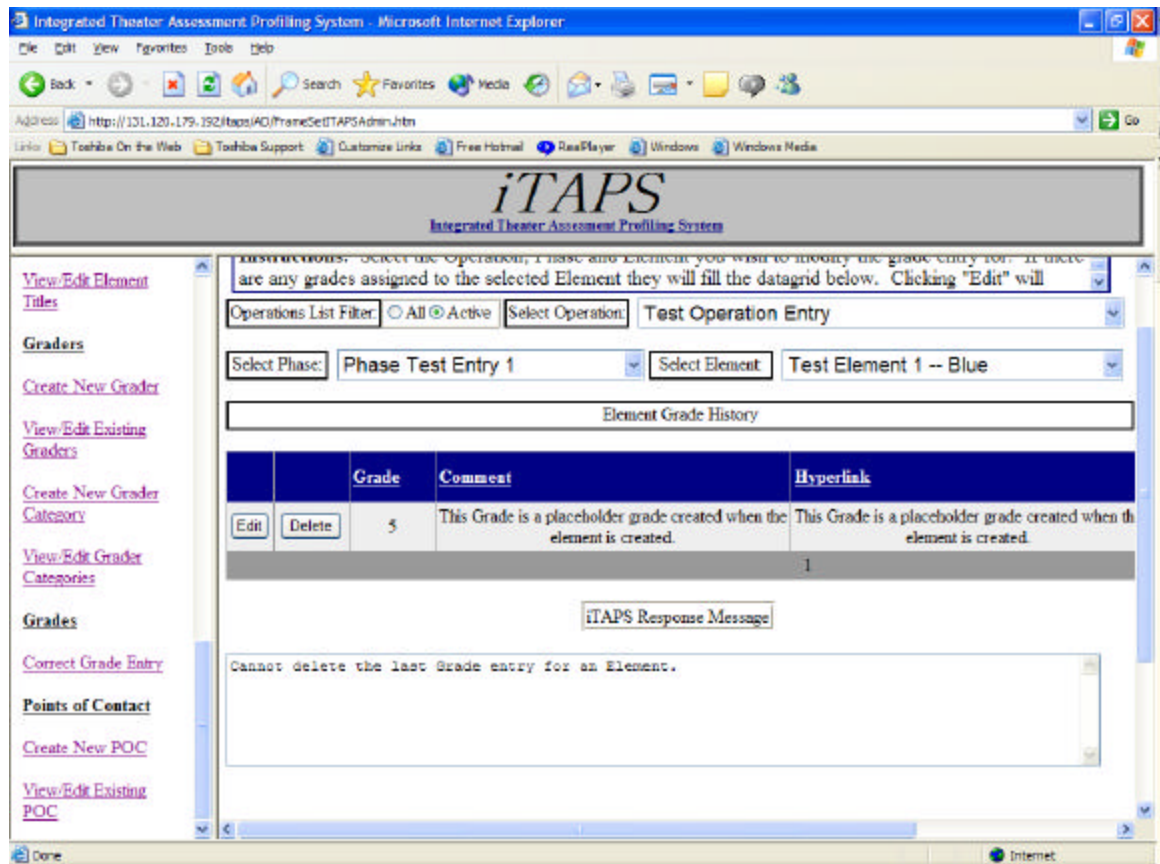


Figure 91. ITAPS Responds

After the deletion, ITAPS will list the results of the operation in the Response Message window (see figure 91). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.19 Duplicating an Operation Structure

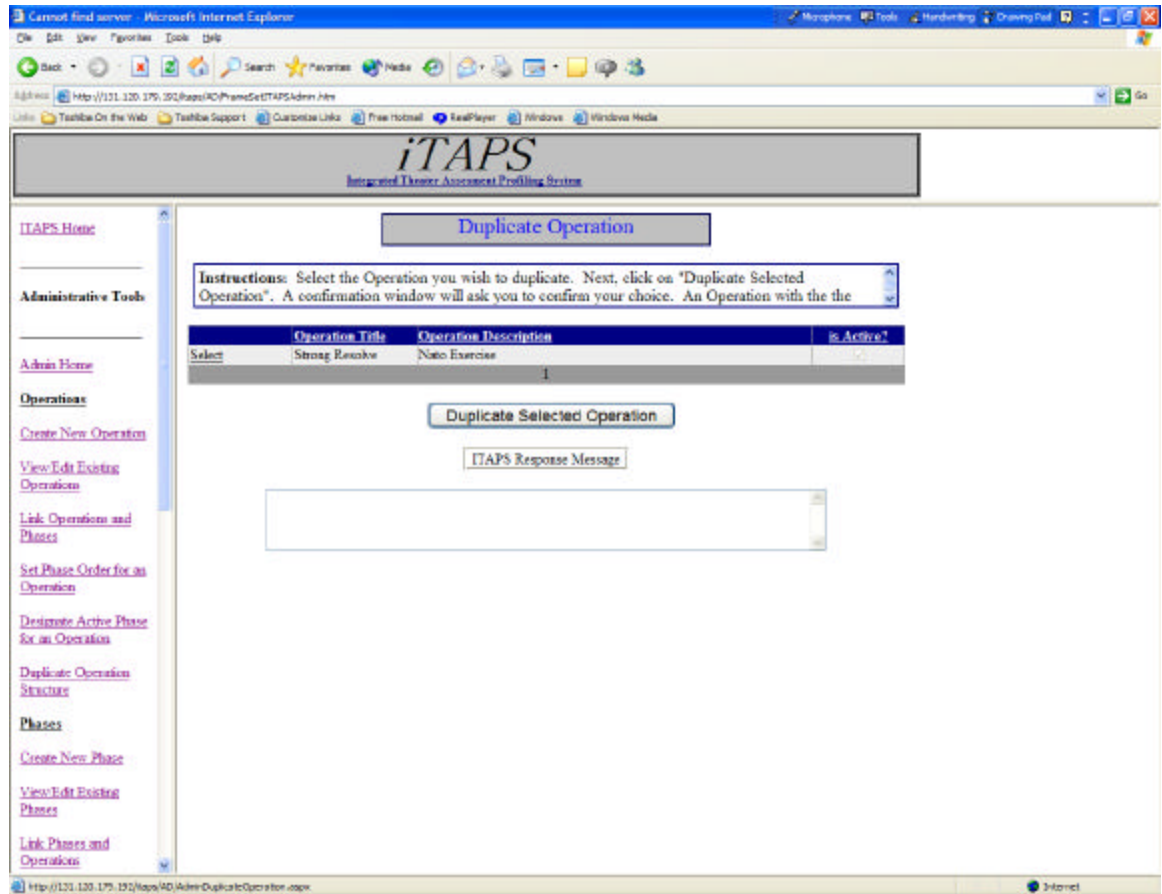


Figure 91a. Duplicate Operation

The Duplicate Operation Page is a tool of convenience to all the Admin User to easily duplicate an Operation's structure to reuse in a new, but similar Operation(such as Exercises). It saves the user from having to manually recreate the Operation structure. To duplicate the Operation, select the Operation you wish to duplicate from the list and click "Duplicate Selected Operation".

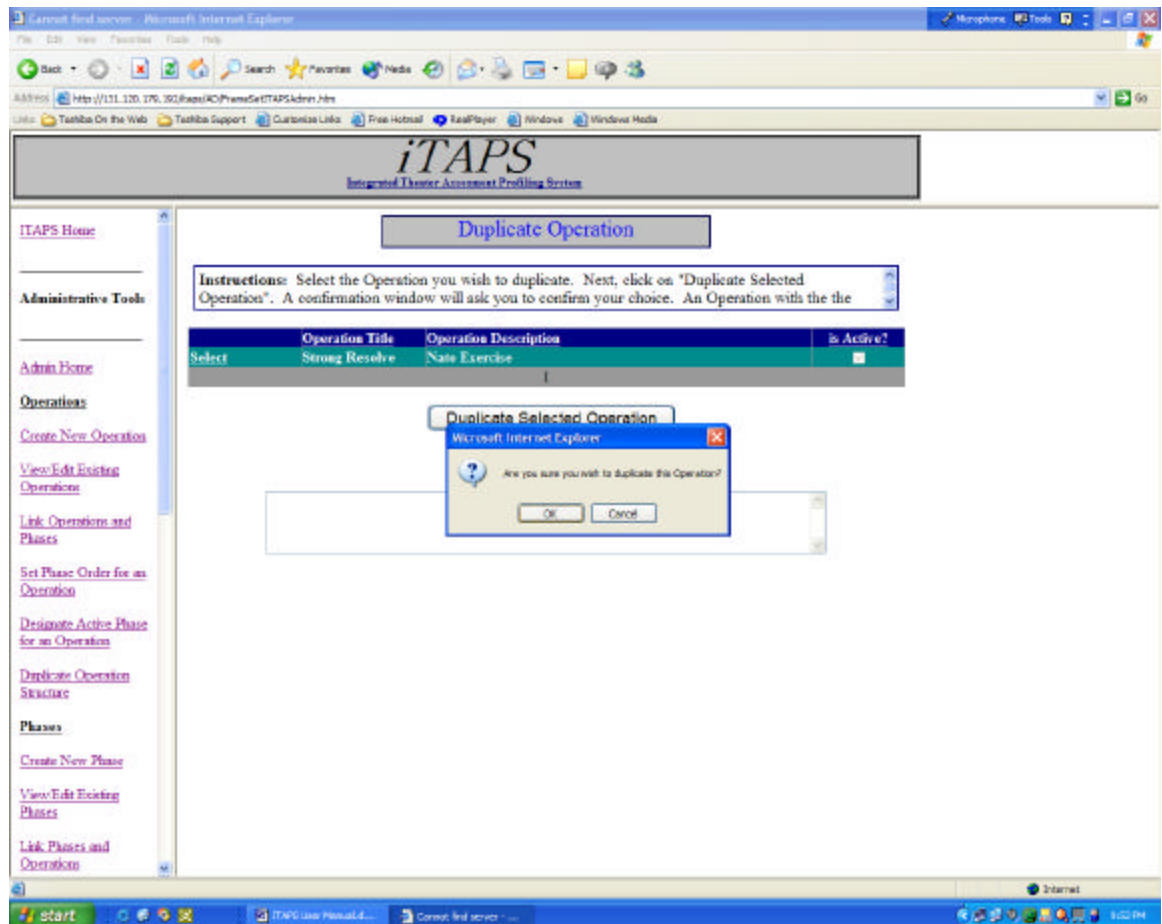


Figure 91b. Confirmation

ITAPS will confirm your request. Click "OK" to duplicate the Operation, click "Cancel" to cancel your request.

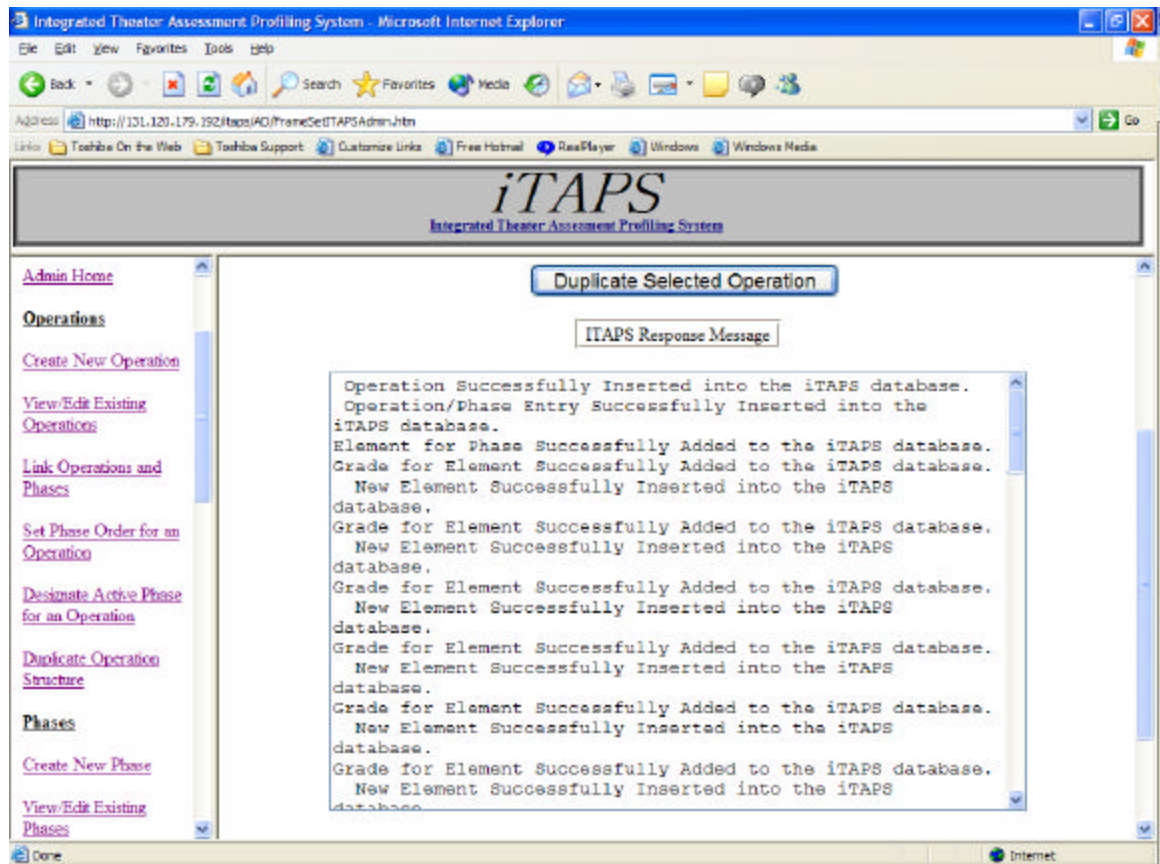


Figure 91c. ITAPS Responds

After the duplication, ITAPS will list the results of the operation in the Response Message window (see figure 91). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.1.20 Side Color Element Duplication

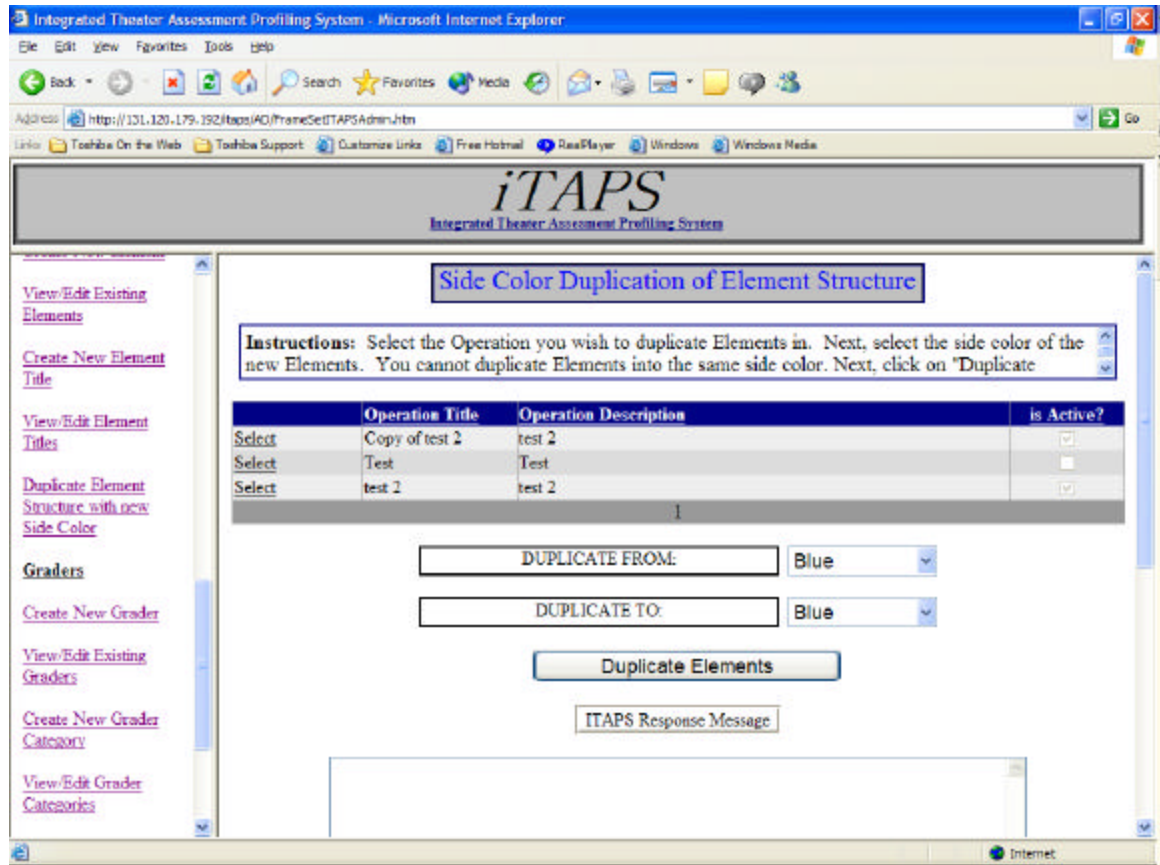


Figure 91d. Duplicating Element Structure with new Side Color

Element Structure duplication is a tool of convenience for the user to easily add another side color with the same Element information without have to manually create all the new Elements for that side color. Select the Operation you wish to duplicate the Element structure in a new side color. Now select the "Duplicate from" color and the "Duplicate To" color and press "Duplicate Elements". ITAPS will not allow duplication to the same color, nor will it allow duplication to a side color which already has elements in the selected Operation with that side color.

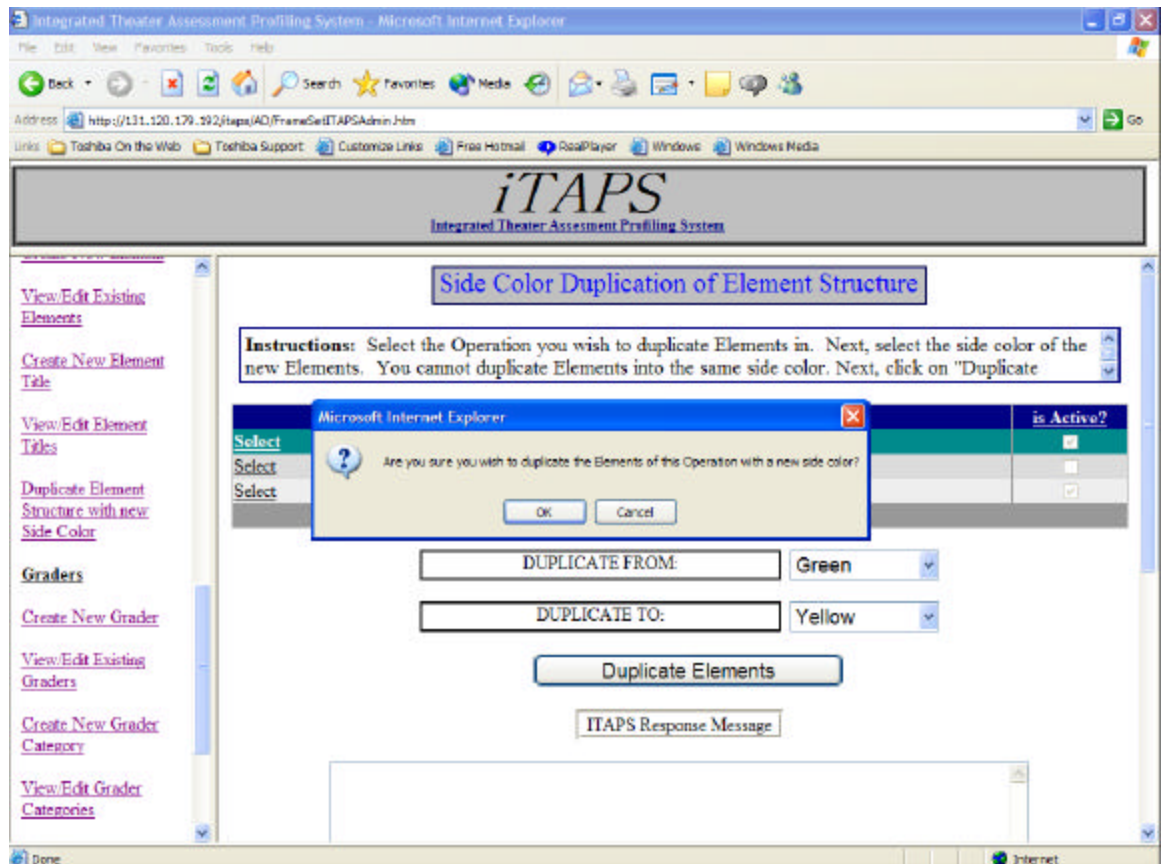


Figure 91e. Selection Confirmation

ITAPS will confirm your request. Click "OK" to duplicate the Elements, click "Cancel" to cancel your request.

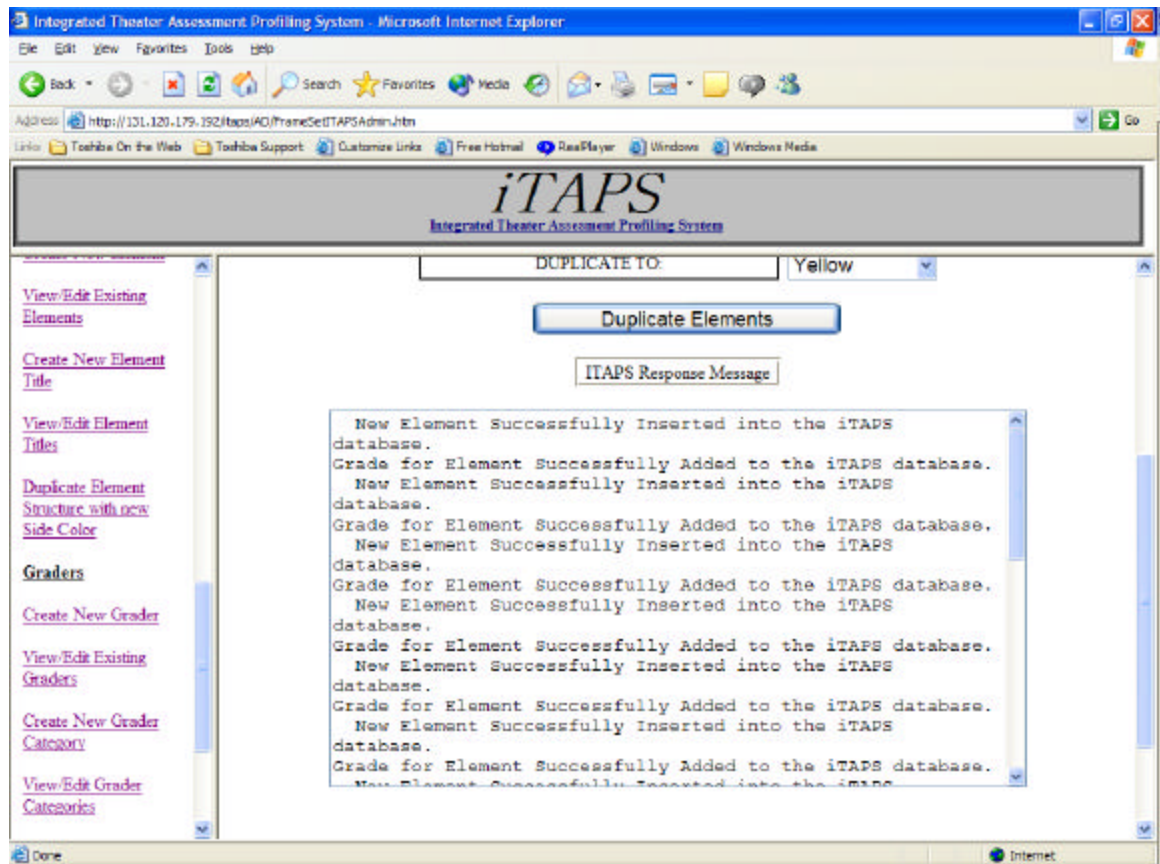


Figure 91f. ITAPS Responds

After the duplication, ITAPS will list the results of the operation in the Response Message window (see figure 91). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.2 Data Input Users

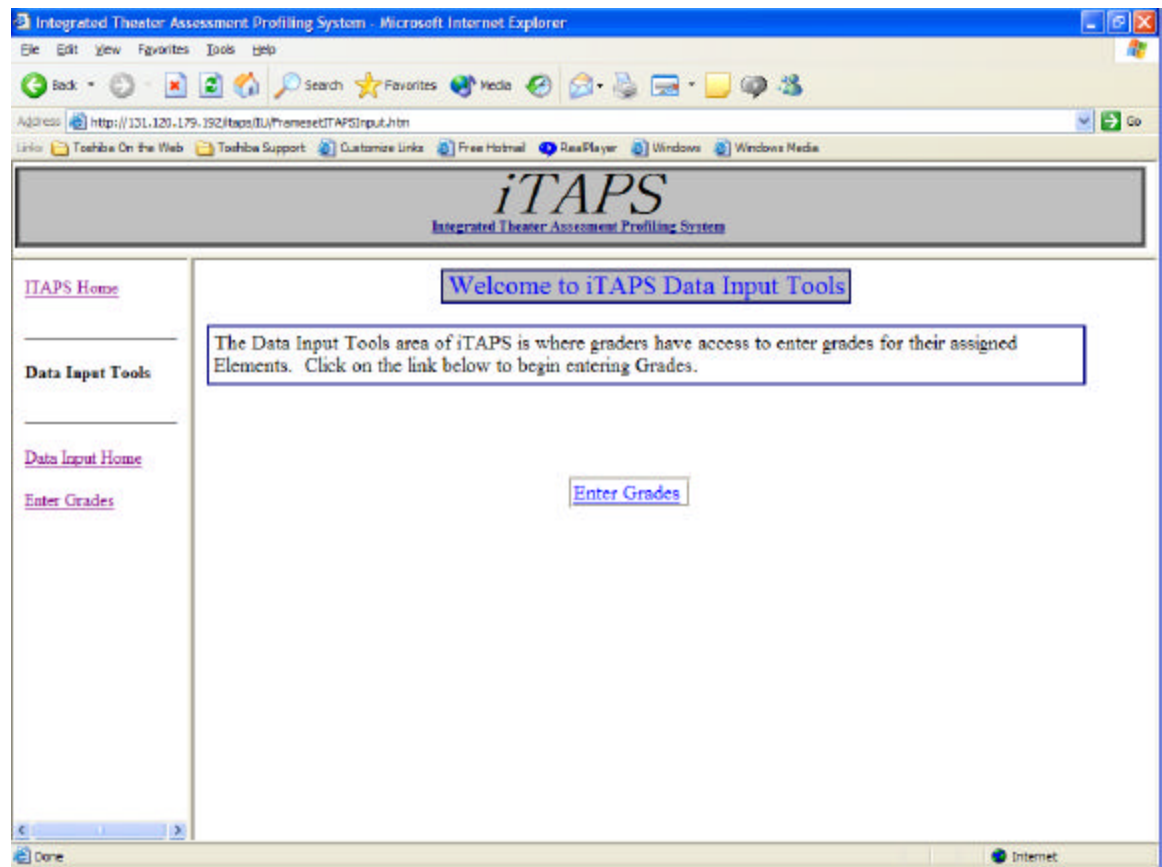


Figure 92. Data Input Tools

The Data Input Tools are used by Graders to update the Element Grades that they are tasked with in an Operation/Phase. The Data Input Tools identify the person using the tools by the Windows User ID and Password they are currently logged in under. Click on “Enter Grades” to make grade entries.

5.2.1 Grade Entry

The screenshot shows a web browser window titled "Integrated Theater Assessment Profiling System - Microsoft Internet Explorer". The address bar shows the URL "http://131.129.179.192/itaps/ITAPhaseSetITAPSInput.htm". The page features a navigation menu on the left with links: "ITAPS Home", "Data Input Tools", "Data Input Home", and "Enter Grades". The main content area is titled "Enter Grades" and contains the following elements:

- Instructions:** ITAPS will identify and authenticate you and display the elements you are authorized to grade. Selected the Element you wish to grade from the drop down list. A grade history for that grade.
- Warning:** If you are not this Grader, please contact your iTAPS Administrator.
- Identification:** You have been identified as the following Grader: **Andy**
- Point-of-Contact:** Your ITAPS Point-of-Contact is: J33A - LCDR Janet Simms - simmsj@secondflt.navy.mil
- Selection:** Select the Operation/Phase/Element you wish to Grade: Test -- Deterrence -- Arms Control
- Form Fields:** Element Description (Test Description), Grade (6), and Comment.
- Grade History Table:**

Grade	Comment	Hyperlink
0	This Grade is a placeholder grade created when the element is created.	This Grade is created when
1		

Figure 93. Entering Grades

The "Enter Grades" page identifies the user currently logged in and fills the Operation/Phase/Element drop down list with only those elements that user is responsible for grading.

Integrated Theater Assessment Profiling System - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Search Favorites Media Print Mail News RSS Feeds

Address http://131.120.179.192/itaps/ITAPHomeSetITAPInput.htm Go

Links Toshiba On the Web Toshiba Support Customize Links Free Hotmail RealPlayer Windows Windows Media

iTAPS

Integrated Theater Assessment Profiling System

[ITAPS Home](#)

[Data Input Tools](#)

[Data Input Home](#)

[Enter Grades](#)

Enter Grades

Instructions: ITAPS will identify and authenticate you and display the elements you are authorized to grade. Selected the Element you wish to grade from the drop down list. A grade history for that grade

If you are not this Grader, please contact your iTAPS Administrator

You have been identified as the following Grader: **Andy**

Your ITAPS Point-of-Contact is: J33A - LCDR Janet Simms - simmsj@secondflt.navy.mil

Select the Operation Phase/Element you wish to Grade: Test -- Deterrence -- Arms Control

Element Description	Test Description

New Grade Entry Panel

Grade: 6

Comment:

Hyperlink:

Grade History

Grade	Comment	Hyperlink
0	This Grade is a placeholder grade created when the element is created.	This Grade is created when...

Figure 94. Entering Grades

Select the Element you wish to grade from the drop down list. Enter Grades, Comments and Hyperlinks(for giving amplifying documentation) .

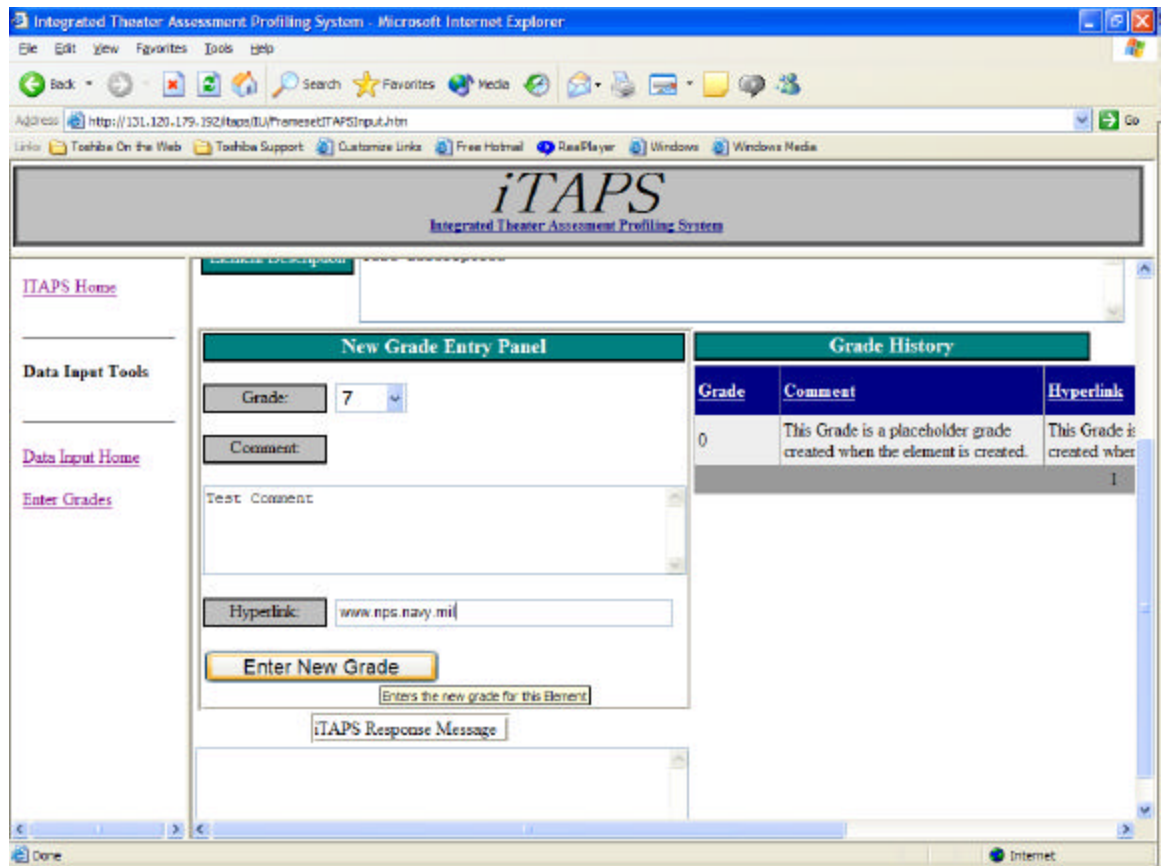


Figure 95. Entering Grades

Once you are satisfied with your entry, click “Enter New Grade”.

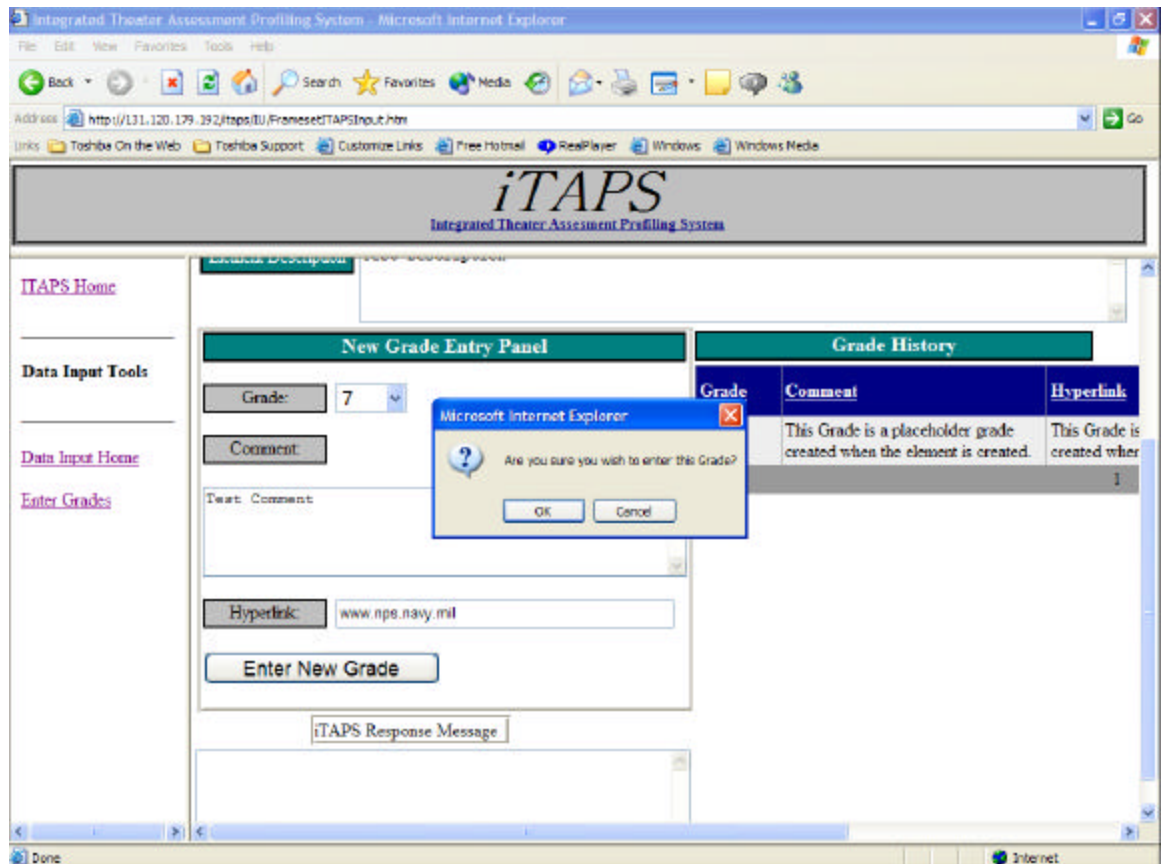


Figure 96. Confirming your entry

A confirmation box will ask you to confirm your selection (see figure 52). Clicking “OK” will cause the Grade to be entered. Clicking “Cancel” will cancel your Grade Entry.

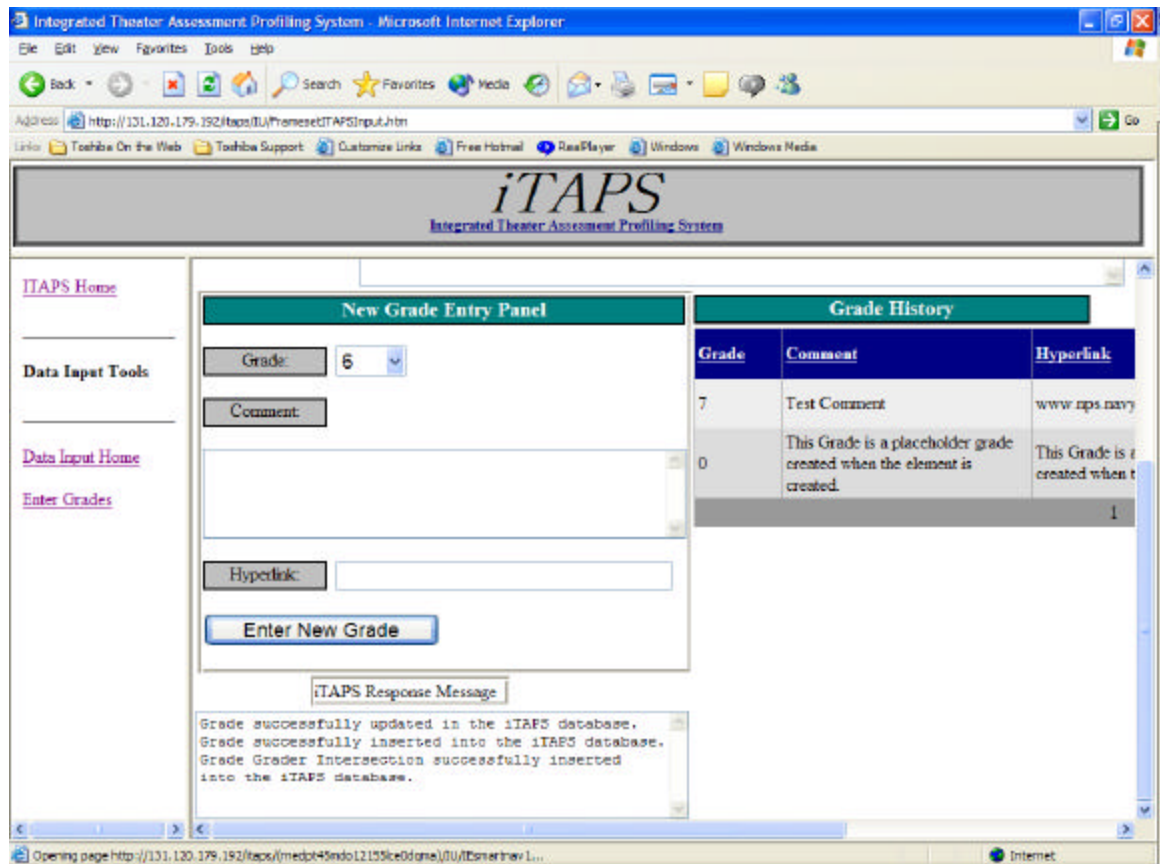


Figure 97. ITAPS Responds

After a Grade insertion, ITAPS will list the results of the operation in the Response Message window (see figure 97). If any error messages result, contact your system administrator and consult the online troubleshooting section of SQL Server as these error messages are SQL Server error messages and indicate a problem with the SQL Server database or the connection to the SQL Server database.

5.3 View Users

View users see the output of ITAPS as a set of radar diagrams that synthesize the input data. By clicking on the node for each element it is possible to drill down to the grades that compose that element. When a “bottom” element is reached, a display of the grade, comments and grader information is displayed.

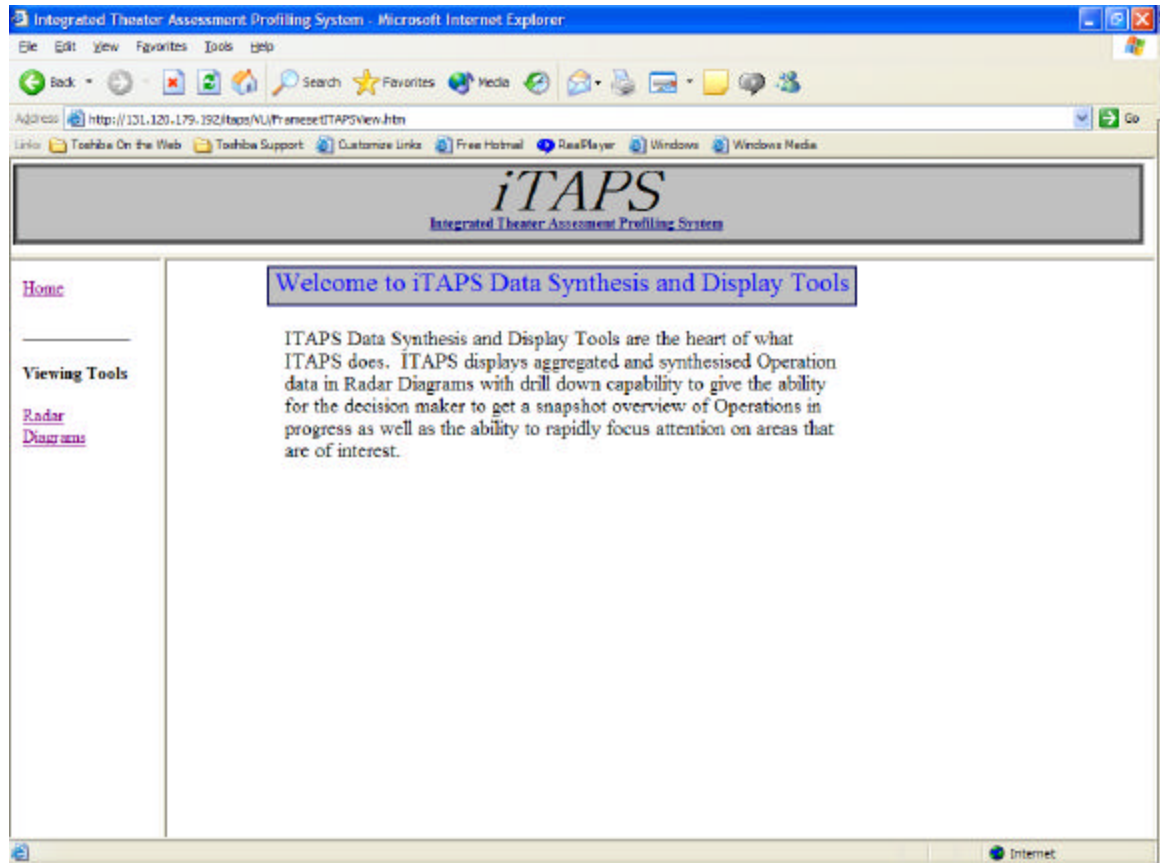


Figure 98. Data Synthesis and Display Tools

Click on “Radar Diagrams” to begin viewing data.

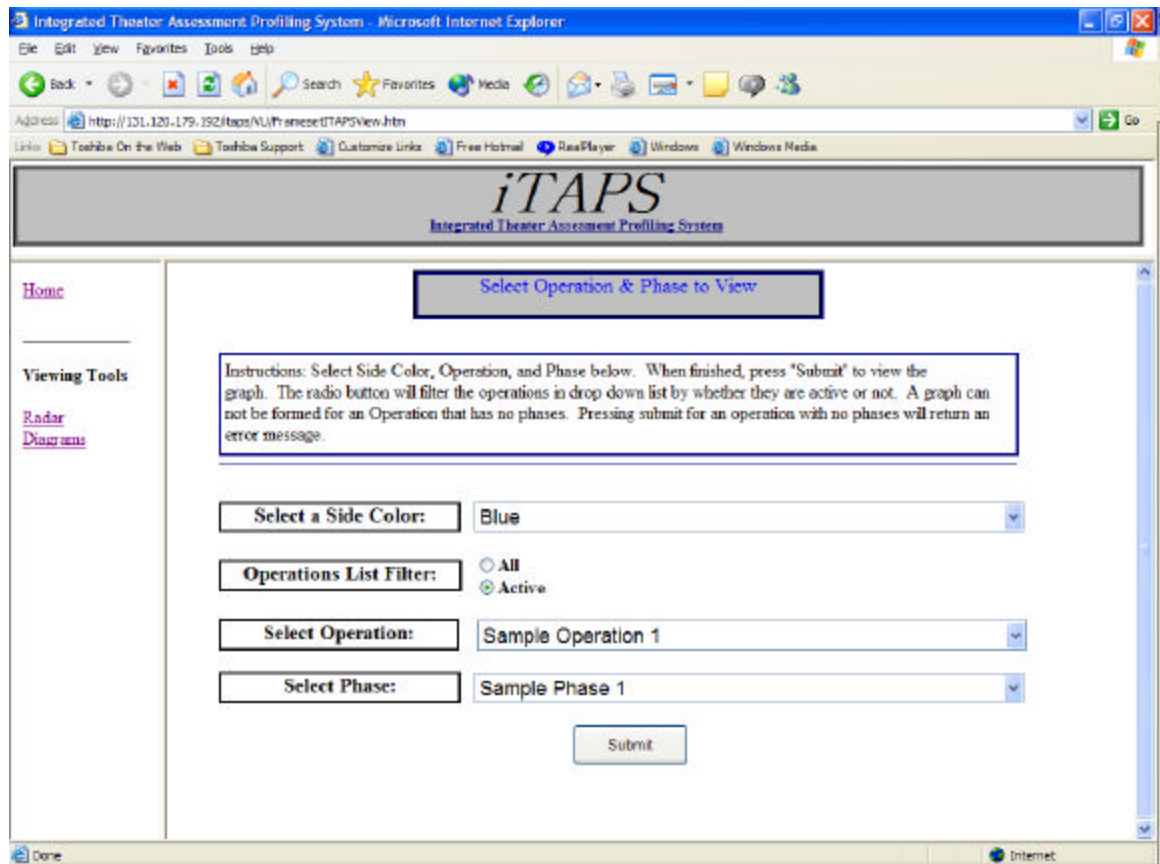


Figure 99. Selecting an Operation and Phase to view

Select the Operation and Phase you wish to examine using the drop down lists. Click “Submit” when done. The entries here are error trapped to ensure proper data display in the radar diagrams.

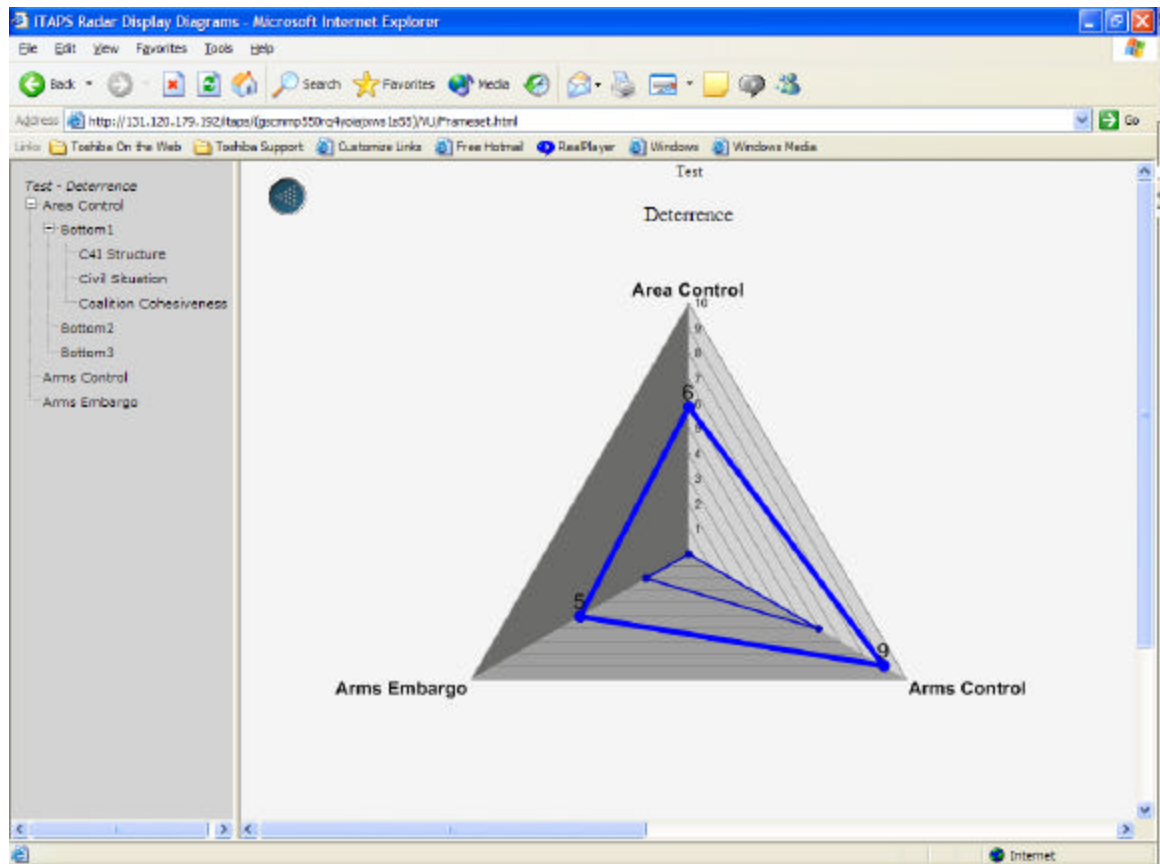


Figure 99. Radar Diagrams

ITAPS now displays the top level radar diagram of the selected Operation and Phase in the right frame and displays a tree diagram of the Operation and Phase in the left frame. The grades are compiled on the fly and are the most current possible. The Grade history is shown by progressively smaller and darker. There are 3 levels of history viewable.

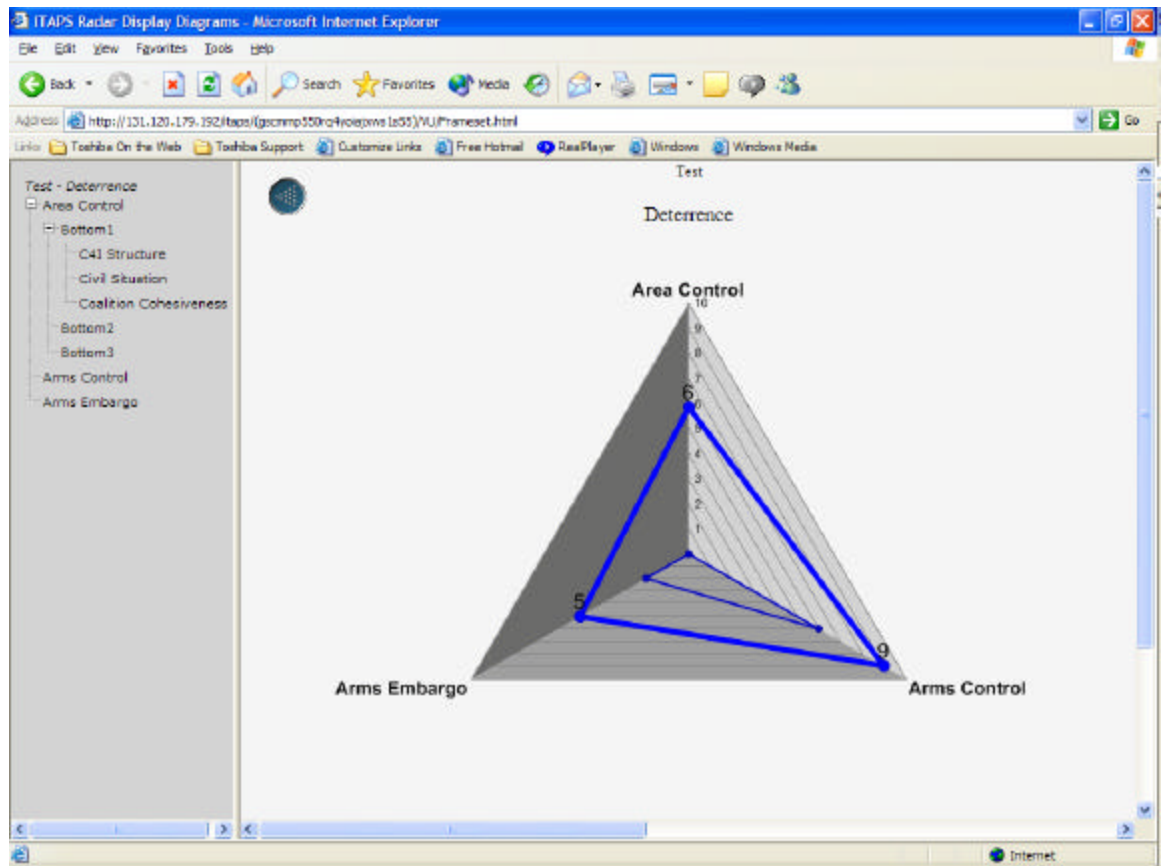


Figure 100. Selecting a Child Element

To drill down to the subordinate Elements you can either click on the appropriate link in the tree on the left or click on the points on the radar diagram.

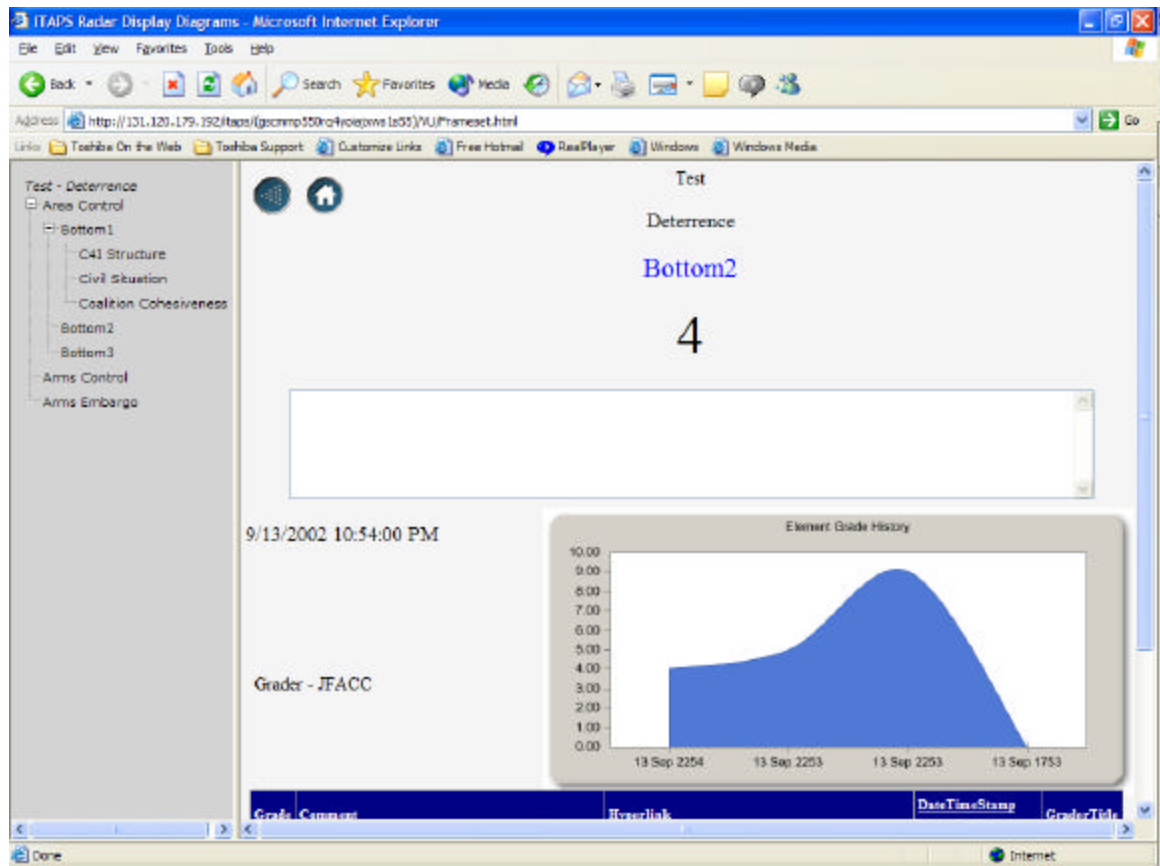


Figure 101. Selecting a Child Element

When you click on a “bottom” element it will display the latest grade that was entered and information associated with that grade. A graph and a datagrid provide grade history.

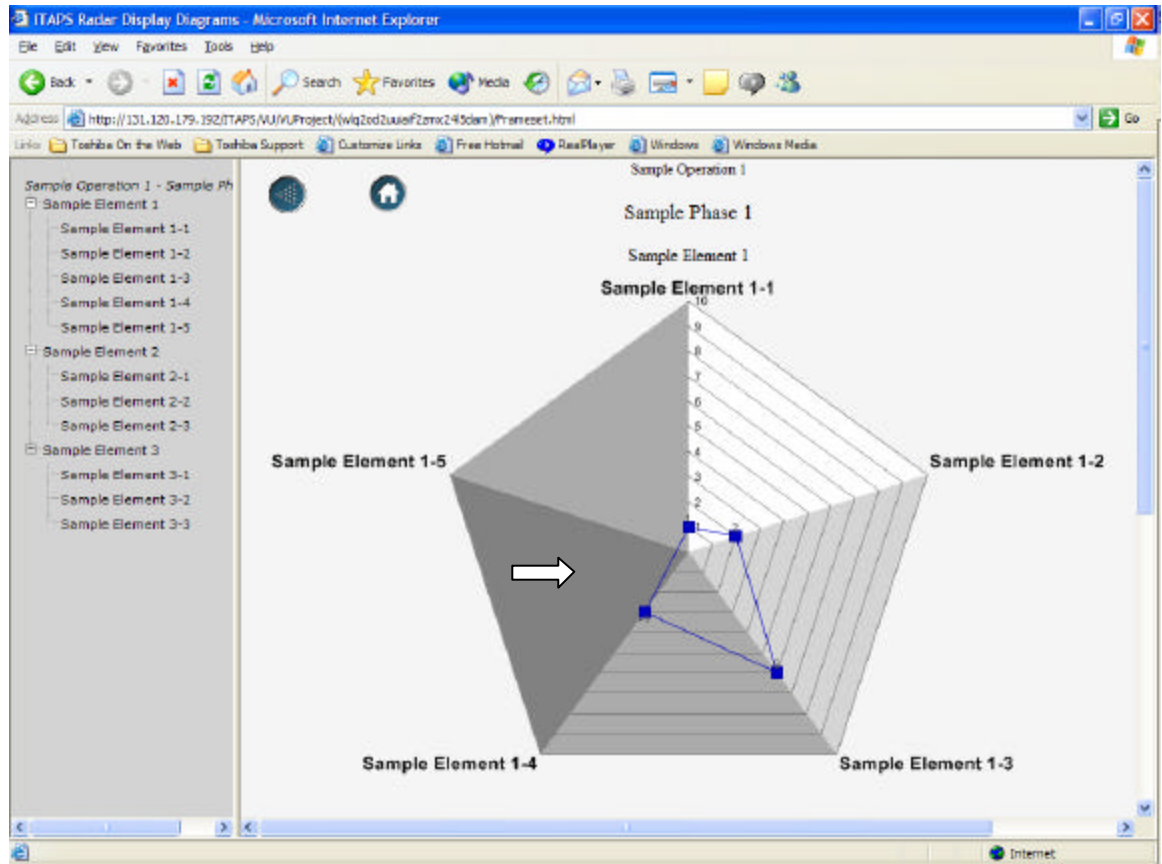


Figure 102. Selecting a Child Element

If grade is a “bottom” Element but is set to “autograded”, this is an error and ITAPS will illustrate this by bypassing the offending grade entry as illustrated in the picture above. This is a feature of the Charting plugin used by ITAPS.

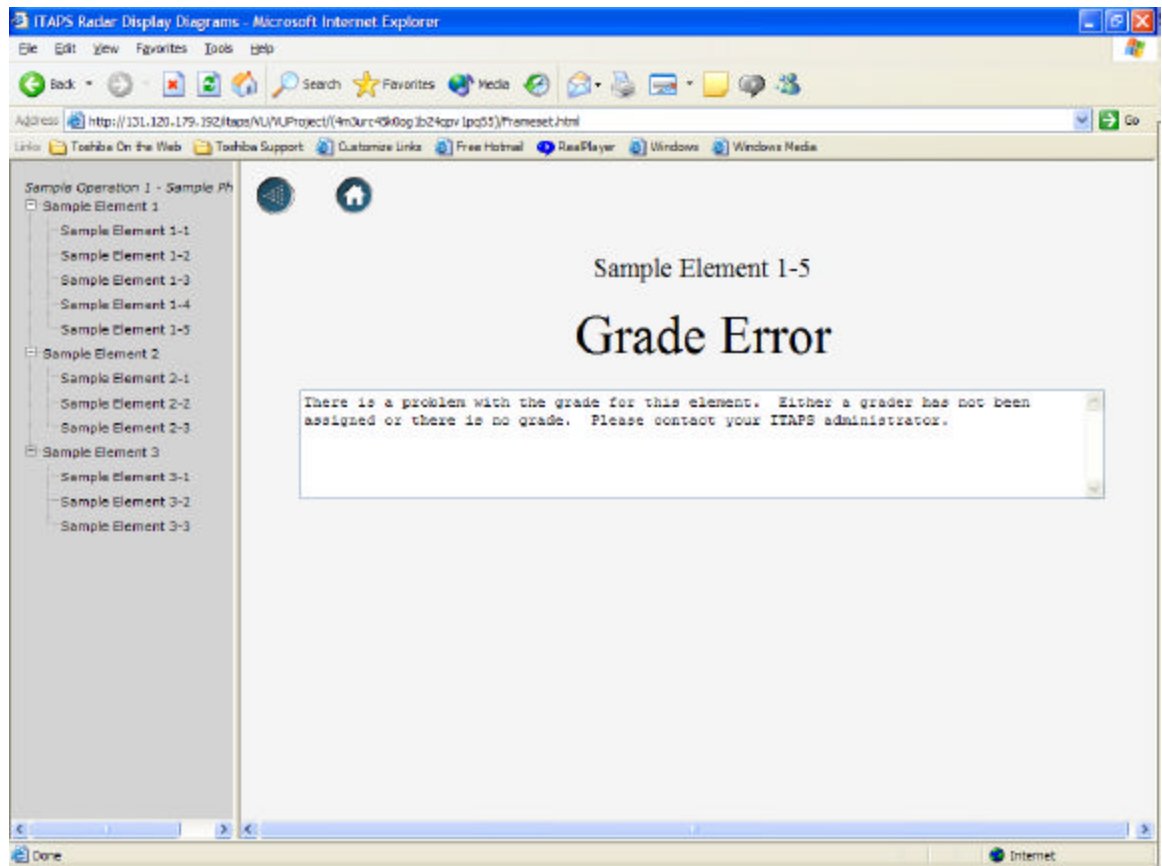


Figure 103. Selecting a Child Element

If the offending Element is selected by using the tree, the above screen is displayed to notify the user of the error.

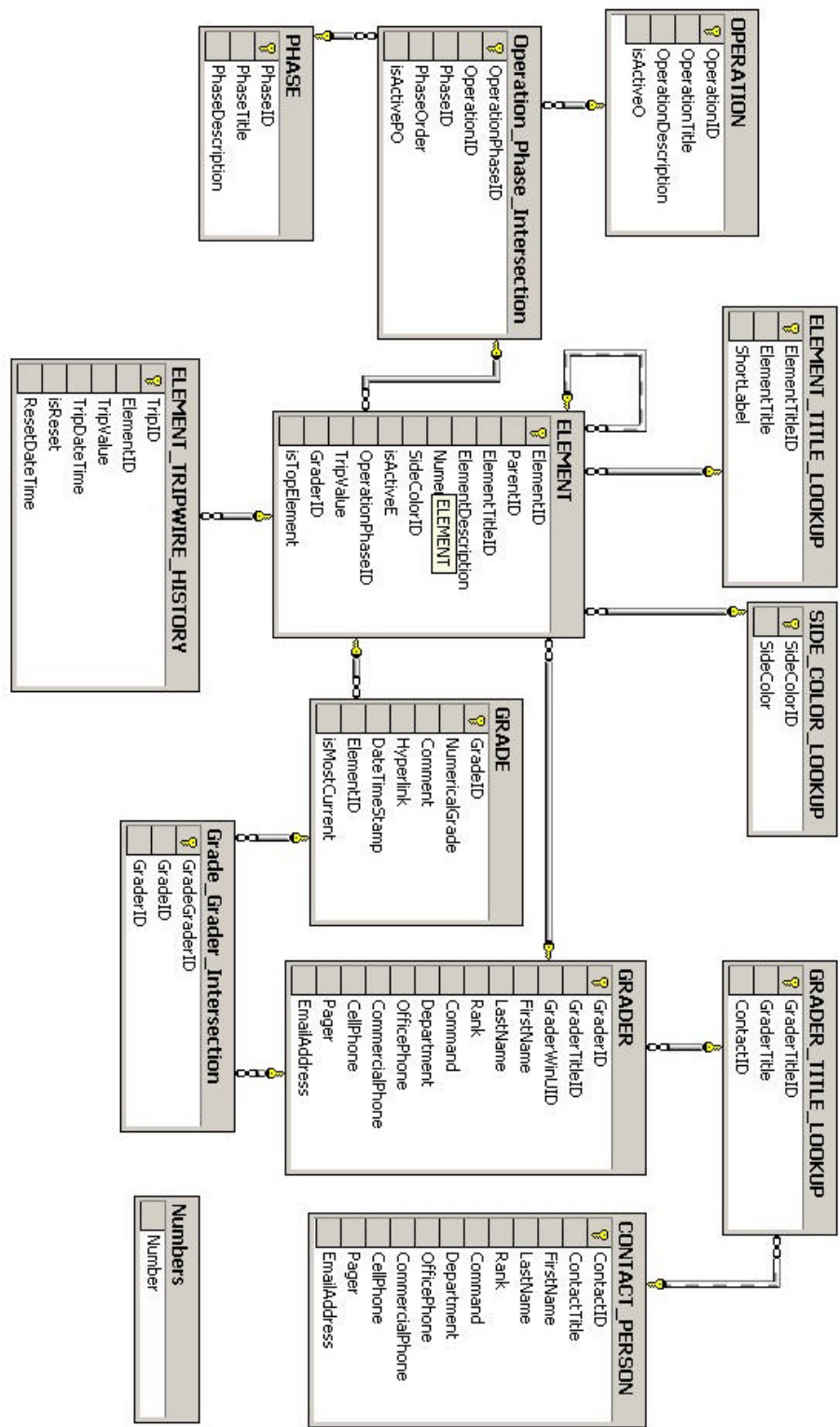
5.4 Hints and Tips

Here are some hints and tips to make your use of the software as error-free as possible.

- Only access the pages via the built-in links within ITAPS. Do not try to manually enter the urls into your browser. Some ITAPS pages depend on data generated in previous pages and can cause errors or unpredictable behavior if not properly accessed.
- Construct your data in the following manner for the optimal “flow” and minimal backtracking:
 - Create an Operation(designate as “not active”)
 - Create and/or link Phases to your Operation
 - Designate the active Phase
 - Create/select POCs for your Grader Categories
 - Create/select Grader Categories you know you will need for your Operation’s Elements
 - Create/select Graders you know you will need for your Operation’s Elements
 - Create the Elements of your Operation and Phases from the top down. (You can check your structure visually by using the View Tools Radar Diagrams and the left pane tree view...NOTE: this will only work after the top 3 elements are created)
- You cannot have two Elements of the same ElementTitle and SideColor in an Operation/ Phase. The charting tool uses the ElementTitle to construct its diagrams and having identical names causes non-deterministic results. ITAPS is coded to prevent the user from inadvertently doing this so when you “View/Edit Elements”, the Element Title drop down list in the Edit view will not include all Element Titles.
- ITAPS is a server-run software solution not a client-side windows application...Give the system time to respond. Rapidly changing multiple items on a page without waiting for dependent items to fill with the proper data will result in non-deterministic results.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B DATABASE SCHEMA



THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Marine Corps Representative
Naval Postgraduate School
Monterey, California
4. Director, Training and Education, MCCDC, Code C46
Quantico, Virginia
5. Director, Marine Corps Research Center, MCCDC, Code C40RC
Quantico, Virginia
6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
Camp Pendleton, California
7. Director, Studies and Analysis Division, MCCDC, Code C45
Quantico, Virginia
8. J6A Commander Second Fleet
Norfolk, Virginia
9. J65 Commander Second Fleet
Norfolk, Virginia

THIS PAGE INTENTIONALLY LEFT BLANK